IN THE UNITED STATES DISTRICT COURT
FOR THE WESTERN DISTRICT OF TEXAS
WACO DIVISION

AdaptFlow Technologies LLC, Plaintiff, v. **Roku, Inc.,** Defendant.	**Case No. 6:22-cv-00996** **Jury Trial Demanded**

COMPLAINT FOR PATENT INFRINGEMENT

Plaintiff AdaptFlow Technologies LLC ("AdaptFlow") files this Complaint against Roku, Inc. for patent infringement of United States Patent Nos. 7,120,034; 7,206,494; 7,340,528; 7,539,012; 8,447,162; 8,667,068; 9,838,757; and 10,015,064 (the "patents-in-suit"), and alleges as follows:

NATURE OF THE ACTION

1. This is an action for patent infringement arising under the patent laws of the United States, 35 U.S.C. §§ 1 *et seq.*

THE PARTIES

2. AdaptFlow is a limited liability company organized under laws of the State of Texas with its principal place of business situated at 10412 China Spring Rd. Ste F PMB 1033, Waco, TX 76708.

3. On information and belief, defendant Roku, Inc. ("Roku") is a corporation organized under the laws of the State of Delaware with a principal place of business at 1155 Coleman Avenue, San Jose, California 95110.

4. Roku maintains offices within this State and judicial district at 9606 North Mopac Expressway, Suite 400, Austin, Texas 78759.

5. On information and belief, Roku sells and offers to sell products and services throughout Texas, including in this judicial district, as well as throughout the United States, and introduces products and services that perform infringing processes into the stream of commerce knowing that they would be used, offered for sale, or sold in this judicial district and elsewhere in the United States.

6. On information and belief, Roku has made, used, offered to sell and/or sold products and services, including the following specifically accused products and services: (1) Roku TV; (2) Roku Streaming Players; (3) Roku Ultra; (4) the Roku Platform; (5) current or legacy products or services, which use, or have used, one or more of the foregoing products and services as a component product or component service; (6) combinations of products and/or services comprising, in whole or in part, two or more of the foregoing products and services; and (7) all other current or legacy products and services imported, made, used, sold, or offered for sale by Roku that

operate, or have operated in a substantially similar manner as the above-listed products and services. (As used herein, one or more of the forgoing products and services are individually and collectively referred to as the accused "Roku Products and Services").

7. On information and belief, Roku, as well as the hardware and software components comprising the Roku Products and Services and/or that enable the Roku Products and Services to operate, including but not limited to servers, server software, streaming player hardware, streaming player software, streaming player firmware, Roku Advertising Framework software, Roku closed captioning software, Roku parental control software, Roku interactive advertising software, Roku content delivery software, Roku content delivery hardware (individually and collectively referred to herein as the accused "Roku System"), infringes (literally and/or under the doctrine of equivalents) at least one claim of each of the patents-in-suit.

JURISDICTION AND VENUE

8. This Court has personal jurisdiction over Roku because it committed and continues to commit acts of infringement in this judicial district in violation of 35 U.S.C. §§ 271(a). In particular, on information and belief, Roku has made, used, offered to sell access to, and/or sold access to the accused Roku Products and Services in the Western District of Texas, and has made, used, offered to sell access to, and/or sold access to the Roku System in the Western District of Texas.

9. On information and belief, Roku is subject to the Court's jurisdiction because it regularly conducts and solicits business, or otherwise engages in other persistent courses of conduct in this judicial district, and/or derives substantial revenue

from the use, sale, and distribution of goods and services, including but not limited to the accused Roku Products and Services provided to individuals and businesses in the Western District of Texas.

10. On information and belief, Roku infringes the patent-in-suit in Texas, including specifically the Western District of Texas, at least, by making, using, offering to sell access to, and/or selling access to the accused Roku Products and Services in the Western District of Texas, and its making, use, offering to sell access to, and/or selling access to the Roku System.

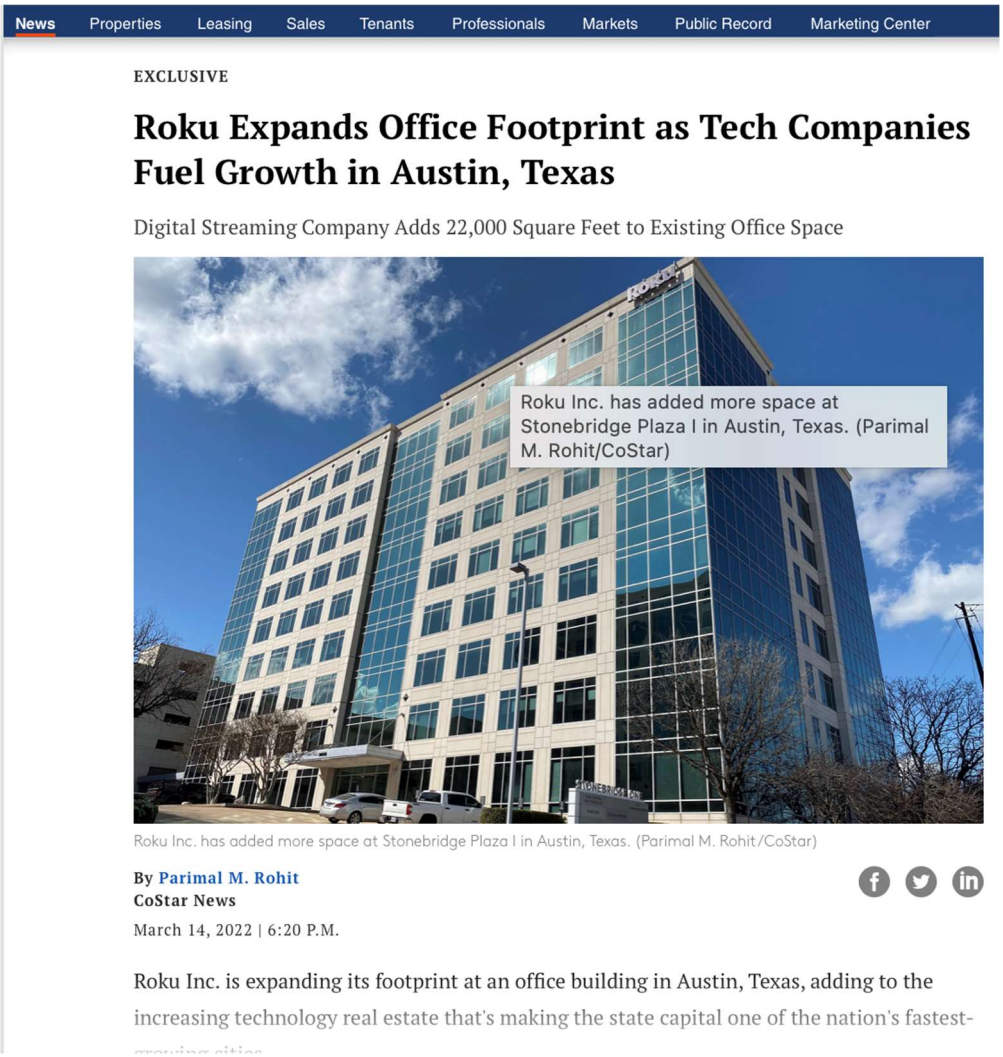
11. Roku markets itself as the "No. 1 platform for streaming TV in the U.S." with a mission "to be the streaming platform that connects and benefits the entire TV ecosystem around the world." *See* https://www.weareroku.com/home and https://www.roku.com/about/company. Roku "connect[s] users to the streaming content they love, . . . enable[s] content publishers to build and monetize large

audiences, and . . . provide[s] advertisers with sophisticated tools to reach and engage consumers." *Id.*

12. Upon information and belief, Roku streaming devices are used by millions of consumers in North America, Latin America, and parts of Europe. *Id.*

13. The Roku Products and Services are available throughout this judicial district and throughout the State of Texas.

14. Upon information and belief, Roku employees in Austin support Roku's provision of the Roku Products and Services.



News Properties Leasing Sales Tenants Professionals Markets Public Record Marketing Center

EXCLUSIVE

Roku Expands Office Footprint as Tech Companies Fuel Growth in Austin, Texas

Digital Streaming Company Adds 22,000 Square Feet to Existing Office Space

Roku Inc. has added more space at Stonebridge Plaza I in Austin, Texas. (Parimal M. Rohit/CoStar)

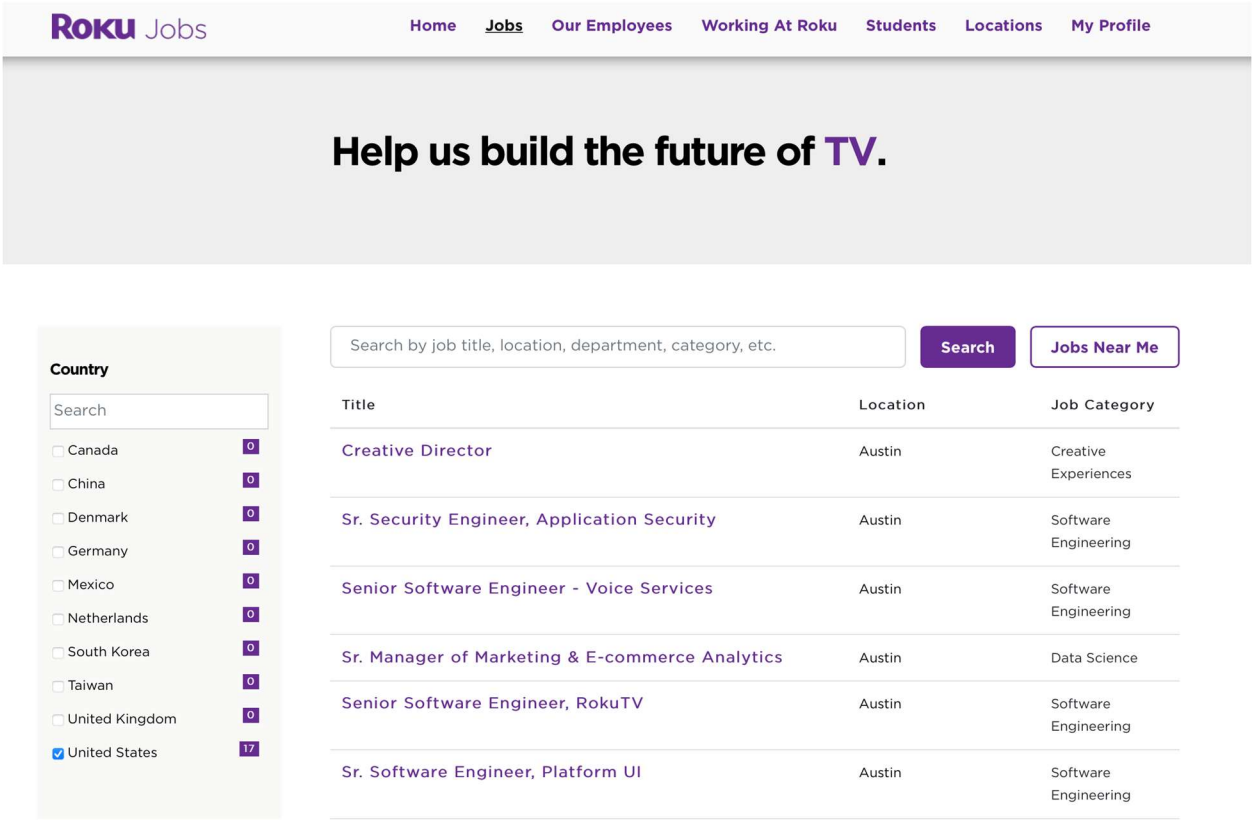
Roku Inc. has added more space at Stonebridge Plaza I in Austin, Texas. (Parimal M. Rohit/CoStar)

By Parimal M. Rohit
CoStar News
March 14, 2022 | 6:20 P.M.

Roku Inc. is expanding its footprint at an office building in Austin, Texas, adding to the increasing technology real estate that's making the state capital one of the nation's fastest-

See https://www.costar.com/article/759037989/roku-expands-office-footprint-as-tech-companies-fuel-growth-in-austin-texas.

15. In addition, Roku continues to grow its presence in this State and District as shown by the job postings below:



See https://www.weareroku.com/jobs/search?page=1&cities%5B%5D=Austin&query=.

16. On information and belief, the accused Roku Products and Services and/or the Roku System are made, used, sold and offered for sale by Roku, its subsidiaries and/or agents, in the Western District of Texas.

17. On information and belief, Roku customers located in the Western District of Texas have obtained access to and used the accused Roku Products and Services and/or the Roku System while located in the Western District of Texas.

18. The Court has personal jurisdiction over Roku at least because it has continuous business contacts in the State of Texas and in the Western District of Texas and Roku has engaged in business activities including transacting business in the Western District of Texas and purposefully directing its business activities, including the sale or offer for sale of the Roku Products and Services to the Western District of

Texas to aid, abet, or contribute to the infringement of third parties in the Western District of Texas.

19. This Court has personal jurisdiction over Roku because, *inter alia*, Roku, on information and belief: (1) has committed acts of patent infringement in this Western District of Texas; (2) maintains a regular and established place of business in the Western District of Texas; (3) has substantial, continuous, and systematic contacts with this State and the Western District of Texas; (4) owns, manages, and operates facilities in this State and the Western District of Texas; (5) enjoys substantial income from its operations and sales in this State and the Western District of Texas; (6) employs Texas residents in this State and the Western District of Texas, and (7) solicits business using the Roku Products and Services and Roku System in this State and the Western District of Texas.

20. Venue is proper pursuant to 28 U.S.C. §§ 1391(b), (c), (d) and/or 1400(b), at least because Roku, has committed acts of infringement in this judicial district, and has regular and established places of business in this judicial district.

United States Patent No. 7,120,034

21. On October 10, 2006, the United States Patent and Trademark Office ("USPTO") duly and legally issued United States Patent No. ("the '034 patent") entitled "Method and Apparatus for Reducing Electromagnetic Radiation" to inventors, Brian Jonathan Cromarty, Lawrence Charles Coan, and Edward Allen Hall.

22. The '034 patent is presumed valid under 35 U.S.C. § 282.

23. AdaptFlow owns all rights, title, and interest in the '034 patent.

24. AdaptFlow has not granted Roku an approval, an authorization, or a license to the rights under the '034 patent.

25. The '034 patent relates to, among other things, the electromagnetic isolation of processed signals prior to communicating the signals to a subsequent device.

26. The claimed invention(s) of the '034 patent sought to solve problems with, and improve upon, signal processing systems. For example, the '034 patent states:

> Electromagnetic radiation (EMR) is emitted by every operating electric and/or electronic device. The power of EMR emission varies depending on the size and electrical strength of the device and the electrical current it carries or employs. In addition, unwanted currents on input and output cables to and from an electronic device may cause the cables themselves to become sources of undesirable EMR. EMR is problematic since it may interfere with components of electronic devices located in the vicinity of the source of EMR, causing electromagnetic interference (EMI). It is known that electronic components or products and devices in the vicinity of EMR sources may malfunction if subjected to certain levels of EMI. EMI generated and received by electronic devices must be attenuated not only to keep the devices functioning properly but also to bring the devices into compliance with applicable governmental regulations. In the United States, the Federal Communications Commission (FCC) promulgates regulations to limit the amount of radiation emitted from electronic equipment, and to provide for routine testing of manufactured products to ensure compliance. Similar administrative bodies perform the same function in Canada, Europe and other parts of the world.
>
> One known solution used to contain or reduce unwanted EMR from products containing electronic devices is through the use of shielded cables. Unfortunately, shielded cables are expensive compared to unshielded cables. Shielded cables also contain a larger form factor, thus occupying much more space within a product enclosure. In addition, the integrity of shielded cables can be more easily compromised as the cables are flexed or moved because they are not as flexible as unshielded cables and the use of unshielded cables facilitates ease in manufacturing.

> Another solution to contain or reduce unwanted EMR from products containing electronic devices is to place the equipment in encompassing metal enclosures. The encompassing conductive enclosure forms a Faraday shield which reduces radiated power by forcing the electric field component of an electromagnetic wave to be nearly zero at the surface of the conductor, thereby blocking wave propagation. Unfortunately, emissions through necessary openings in the enclosures for outputting signals, lessens the effectiveness of the enclosures. Additionally, the resulting package is bulky and relatively expensive.

See '034 patent at col. 1, ll. 19-61.

27. The invention(s) claimed in the '034 patent solves various technological problems inherent in prior-art signal processing systems and enables such systems to, among other things: (1) function more efficiently; (2) reduce electromagnetic interference ; (3) have smaller and/or less expensive form factors.

United States Patent No. 7,206,494

28. On April 17, 2007, the USPTO duly and legally issued United States Patent No. ("the '494 patent") entitled "Detection Rules for a Digital Video Recorder" to inventors Joseph C. Engle and Akinyemi Odutola.

29. The '494 patent is presumed valid under 35 U.S.C. § 282.

30. AdaptFlow owns all rights, title, and interest in the '494 patent.

31. AdaptFlow has not granted Roku an approval, an authorization, or a license to the rights under the '494 patent.

32. The '494 patent relates to, among other things, a method for identifying commercial messages in a video signal.

33. The invention(s) claimed in the '494 patent solves various technological problems inherent in prior-art video systems and enables such systems to, among other things, function more efficiently.

United States Patent No. 7,340,528

34. On March 4, 2008, the USPTO duly and legally issued United States Patent No. 7,340,528 ("the '528 patent") entitled "Data Referencing System" to inventors Christophe Colas Noblecourt and Setra Rakotomavo.

35. The '528 patent is presumed valid under 35 U.S.C. § 282.

36. AdaptFlow owns all rights, title, and interest in the '528 patent.

37. AdaptFlow has not granted Roku an approval, an authorization, or a license to the rights under the '528 patent.

38. The '528 patent relates to, among other things, digital television transmission systems.

39. The claimed invention(s) of the '528 patent sought to solve problems with, and improve upon, digital television transmission systems. For example, the '528 patent states:

> The advent of digital transmission systems intended primarily for broadcasting television signals, in particular but not exclusively satellite television systems, has opened up the possibility of using such systems for other purposes. One of these is to provide interactivity with the end user. As used herein, the term "digital transmission system" includes any transmission system for transmitting or broadcasting for example primarily audiovisual or multimedia digital data. Whilst the present invention is particularly applicable to a broadcast digital television system, the invention may also be applicable to a fixed telecommunications network for multimedia internet applications, to a closed circuit television, and so on.

> One way of interacting with the user is to run an application on the receiver/decoder through which the television signal is received. The code for the application could be permanently stored in the receiver/decoder. However, this would be rather limiting. Preferably, the receiver/decoder should be able to download the code for a required application. In this way, more variety may be provided, and applications can be updated as required without any action on the part of the user.

See '528 patent at col. 1, ll. 24-45.

40. The invention(s) claimed in the '528 patent solves various technological problems inherent in prior-art digital television transmission systems and enables such systems to, among other things, provide users of such systems with improved interactive functionality.

United States Patent No. 7,539,012

41. On May 26, 2009, the USPTO duly and legally issued United States Patent No. 7,539,012 ("the '012 patent") entitled "Data Storage Medium Read/Write Unit Comprising a Heat Sink" to inventor Claude Chapel.

42. The '012 patent is presumed valid under 35 U.S.C. § 282.

43. AdaptFlow owns all rights, title, and interest in the '012 patent.

44. AdaptFlow has not granted Roku an approval, an authorization, or a license to the rights under the '012 patent.

45. The '012 patent relates to, among other things, data storage systems.

46. The claimed invention(s) of the '012 patent sought to solve problems with, and improve upon, data storage systems. For example, the '012 patent states:

> The invention relates to a data storage medium read/write unit comprising a heat sink.

> Data storage medium read/write units such as disk drives are used in electronic equipment to read data from (and also in general to record data onto) a storage medium, such as an optical medium (optical disk), a magnetic medium (hard disk or magnetic tape or others: for example, magnetooptic or semiconductor storage device).
>
> These read/write units are not exclusively used in electronic equipment for information technology; they are also employed in consumer electronic products or other audio and/or video products (professional video, for example).
> It has already been proposed that these read/write units, and in particular hard-disk drives, be removable from the equipment designed to receive them. In addition to allowing the medium to be interchangeable between two or more systems, this solution allows the read/write unit and/or its medium to be changed without having to change the whole system, and thus allows advantage to be taken, at a lower cost, of the latest advances in technology as regards read/write units and/or data storage media, accompanied in general by an increase in the storage capacity.
>
> In order to prevent too large a temperature rise in the read/write unit adversely affecting its operation, the use of a heat sink has already been proposed, as in the Patent Application WO 02/063628 for example.
>
> The subject of the present invention is a solution for mounting the heat sink in the read/write unit that best reconciles the various design constraints, especially as regards cooling efficiency, size and cost (for example, material cost and manufacturing cost of the drive).
>
> The invention thus proposes a data storage medium read/write unit, designed to be installed in an electronic apparatus, comprising an electronic read/write device (or electronic module) and a non-conducting housing attached to the electronic read/write device, in which the housing holds at least one heat sink in direct contact with the electronic read/write device for cooling it.

See '012 patent at col. 1, ll. 4-42.

47. The invention(s) claimed in the '012 patent solves various technological problems inherent in prior-art systems and enables such systems to, among other things: (1) increase cooling efficiently; (2) reduce form factor size; and (3) reduce material and manufacturing costs.

United States Patent No. 8,447,162

48. On May 21, 2013, the USPTO duly and legally issued United States Patent No. 8,447,162 ("the '162 patent") entitled "Saving and Restoring Control Settings for Multimedia Content Receivers" to inventors Mark Alan Schultz and Ronald Douglas Johnson.

49. The '162 patent is presumed valid under 35 U.S.C. § 282.

50. AdaptFlow owns all rights, title, and interest in the '162 patent.

51. AdaptFlow has not granted Roku an approval, an authorization, or a license to the rights under the '162 patent.

52. The '162 patent relates to, among other things, saving and restoring control settings for multimedia content receivers.

53. The claimed invention(s) of the '162 patent sought to solve problems with, and improve upon, media content receiver systems. For example, the '162 patent provides for the saving and restoration of control settings on multimedia content receivers.

United States Patent No. 8,667,068

54. On March 4, 2014, the USPTO duly and legally issued United States Patent No. 8,667,068 ("the '068 patent") entitled "Method and Apparatus for Electronic Message Delivery" to inventors Thomas C. Box and Ana Belen Benitez.

55. The '068 patent is presumed valid under 35 U.S.C. § 282.

56. AdaptFlow owns all rights, title, and interest in the '068 patent.

57. AdaptFlow has not granted Roku an approval, an authorization, or a license to the rights under the '068 patent.

58. The '068 patent relates to, among other things, content delivery systems.

59. The claimed invention(s) of the '068 patent sought to solve problems with, and improve upon, content delivery systems. For example, the '068 patent states:

> The term "digital cinema" refers generally to the creation and/or distribution of motion pictures to cinemas in digital form, in contrast to traditional cinema whose content, including the image, sound track and subtitles, exists in an analog form on motion picture film. Digital cinema projection systems now exist and make use of a variety of technical solutions, rather than a single standardized solution embraced by all. The existence of content in digital form affords the opportunity to provide alternative information along with the original content with little effort. For that reason various organizations, such as the Digital Cinema Initiative ("DCI") and the Society for Motion Picture Engineering ("SMPTE") have devoted much effort towards establishing specifications for digital cinema content distribution.
>
> The DCI specification defines preferred solutions in many areas, such as encoding, distribution, and security. The SMPTE has sought to refine the DCI specification from a technical and implementation perspective. The goal of these specifications is to provide an open architecture that promotes high level and uniform technical performance. The system can provide enhanced sound and image content while providing the theatre patron with a more consistent movie experience. The system also affords potential simplification of production and distribution, delivery, and storage of the movie content.
>
> DCI also permits new business opportunities. The distribution of digital media creates the capability to deliver alternative programming. The delivery system may also permit new and improved advertising capabilities, all of which may enhance the profit capabilities of the businesses involved.
>
> Targeted advertising is a relatively new opportunity that has mainly focused on such areas as electronic billboards and broadcasted cellular phone activities. Neither of these activities offers any real advantages in terms of activities during a presentation, such as a movie, at a venue. None of these forms of delivery take advantage of the mere fact that presence at a theatre or event provides a specific type of marketing opportunity. In addition, the actual movie presentation itself

> provides opportunities for promoting materials, and specifically promoting the material at an appropriate time within the movie or event presentation.
>
> Further, the proliferation and penetration of usage of portable electronic consumer devices is steadily increasing. The ability to deliver electronic messages and content directly to these types of devices at a particular event or presentation is a powerful entertainment and marketing tool. Distribution of electronic messages to consumer devices in an entertainment venue is valuable particularly since the target audience is in the immediate area and has an interest in the content.
>
> Currently, some ancillary information, such as addresses, phone numbers, websites, e-mail addresses may be presented using methods employing broadcast techniques. However, the delivery of this information is, in general, not specifically tied to the presentation of the event with respect to presentation timing or location, potentially limiting its overall effectiveness as an entertainment and marketing tool. Therefore, there is a need for providing content specific to the presentation of an event, such as a movie, while the presentation of the event is occurring.

See ' 068 patent at col. 1 , l. 27- col. 2, l. 17.

60. The invention(s) claimed in the '068 patent solves various technological problems inherent in prior-art content delivery systems and enables such systems to, among other things, improve audience engagement.

United States Patent No. 9,838,757

61. On December 5, 2017, the USPTO duly and legally issued United States Patent No. 9,838,757 ("the '757 patent") entitled "Method to Transmit Video Data in a Data Stream and Associated Metadata" to inventors Louis Chevallier, Lionel Oisel, Francois Le Clerc, and Frederic Lefebvre.

62. The '757 patent is presumed valid under 35 U.S.C. § 282.

63. AdaptFlow owns all rights, title, and interest in the '757 patent.

64. AdaptFlow has not granted Roku an approval, an authorization, or a license to the rights under the '757 patent.

65. The '757 patent relates to, among other things, video data transmission systems.

66. The invention(s) claimed in the '757 patent solves various technological problems inherent in prior-art video transmission systems and enables such systems to, among other things: (1) function more efficiently; and (2) improve video annotation, searching, and/or handling.

United States Patent No. 10,015,064

67. On July 3, 2018, the USPTO duly and legally issued United States Patent No. 10,015,064 ("the '064 patent") entitled "Personalization of Information Content by Monitoring Network Traffic" to inventors Dekai Li, Ashwin Kashyap, and Saurabh Mathur.

68. The '064 patent is presumed valid under 35 U.S.C. § 282.

69. AdaptFlow owns all rights, title, and interest in the '064 patent.

70. AdaptFlow has not granted Roku an approval, an authorization, or a license to the rights under the '064 patent.

71. The '064 patent relates to, among other things, content personalization systems.

72. The claimed invention(s) of the '064 patent sought to solve problems with, and improve upon, content personalization systems and enables such systems to, among other things: function more efficiently.

CLAIMS FOR RELIEF

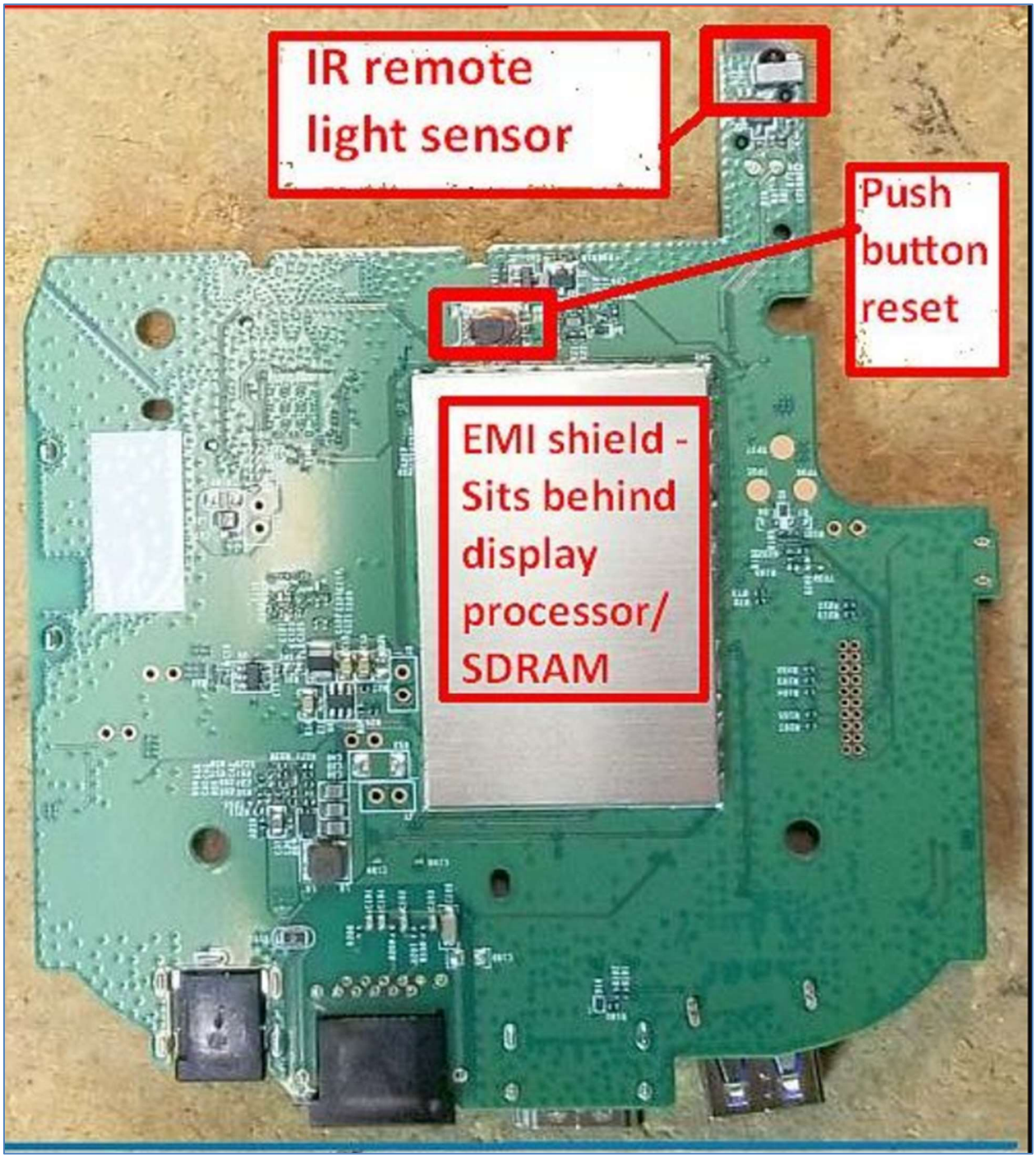
Count I – Infringement of United States Patent No. 7,120,034

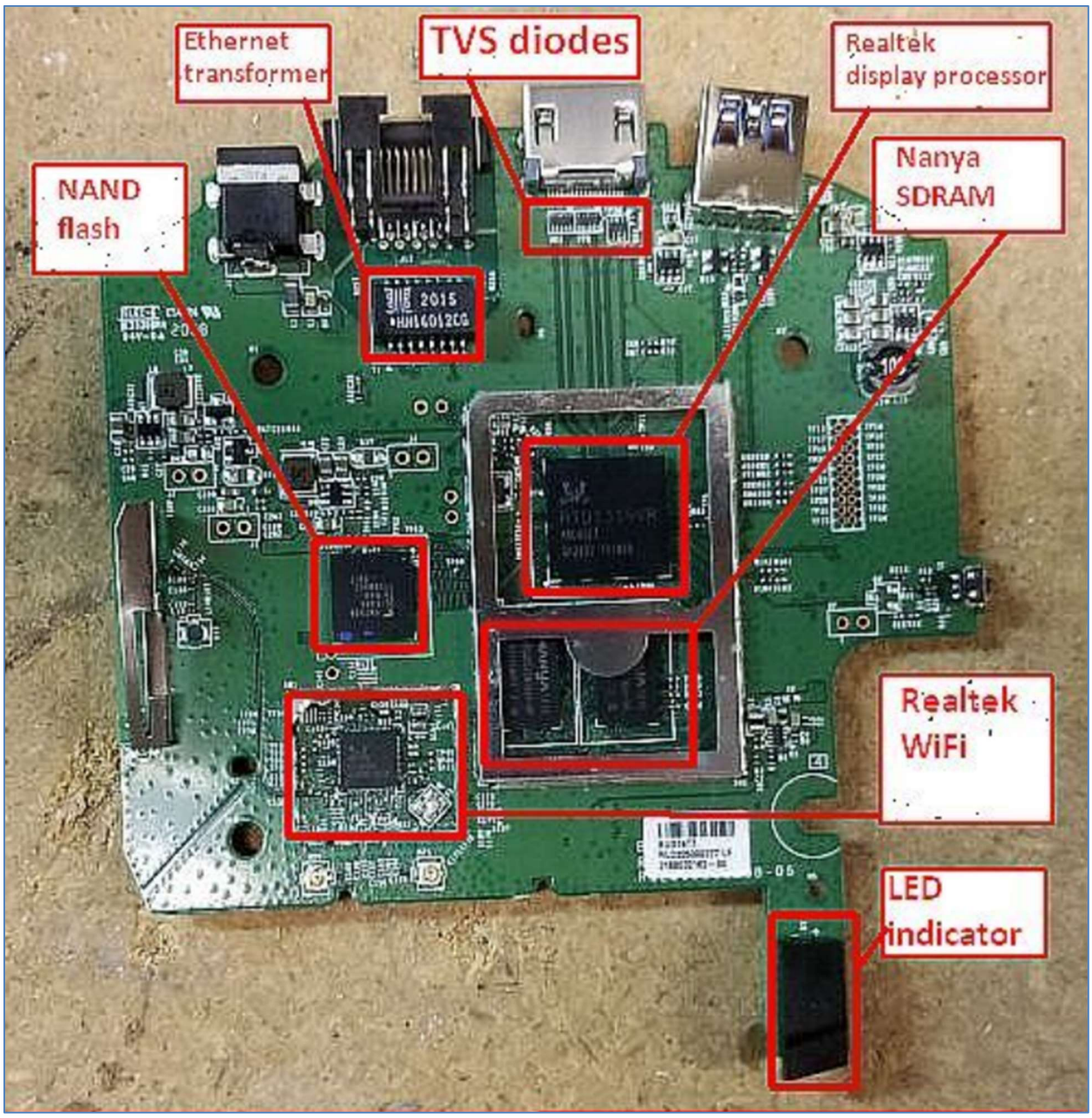
73. AdaptFlow repeats, realleges, and incorporates by reference, as if fully set forth here, the allegations of the preceding paragraphs above.

74. On information and belief, Roku (or those acting on its behalf) makes, uses, sells, imports and/or offers to sell the Roku Products and Services; and makes, uses, sells, sells access to, imports, offers to sell and/or offers to sell access to the Roku System in the United States that infringe (literally and/or under the doctrine of equivalents) at least claim 9 of the '034 patent.

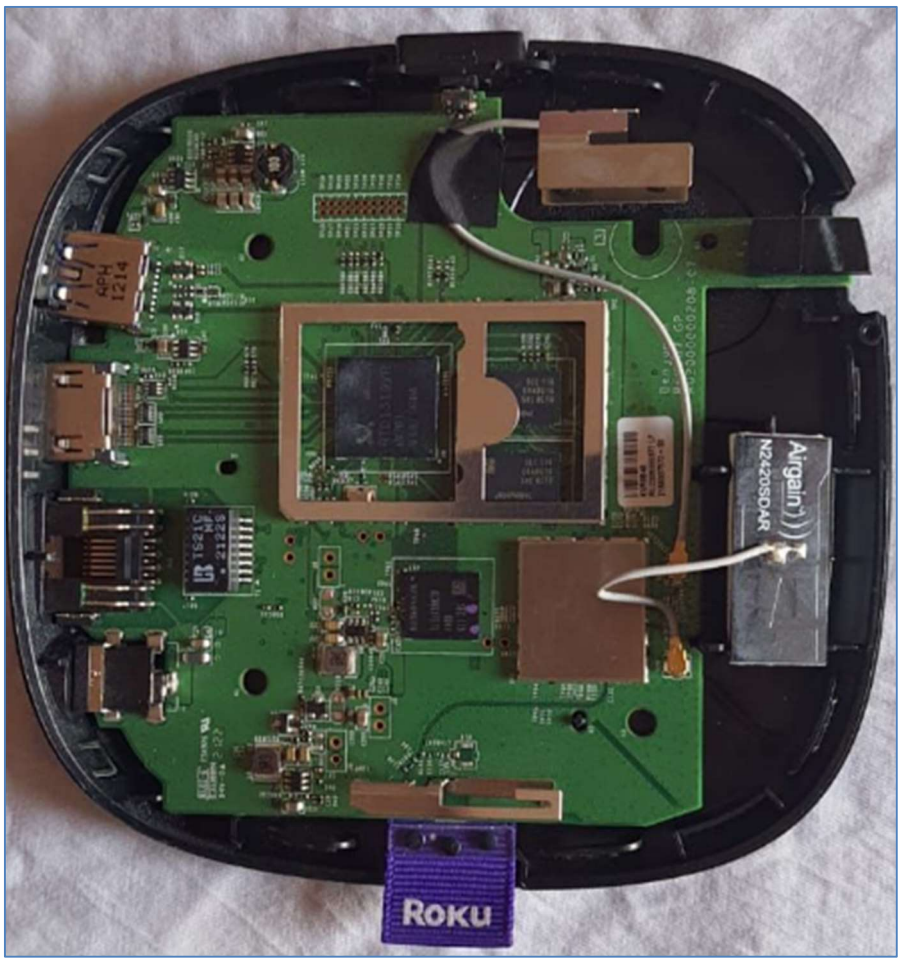
75. On information and belief, one or more components of the Roku system include a printed circuit board for reducing electromagnetic interference, said printed circuit board having an input section and an output section.

See e.g., https://www.roku.com/products/roku-ultra.

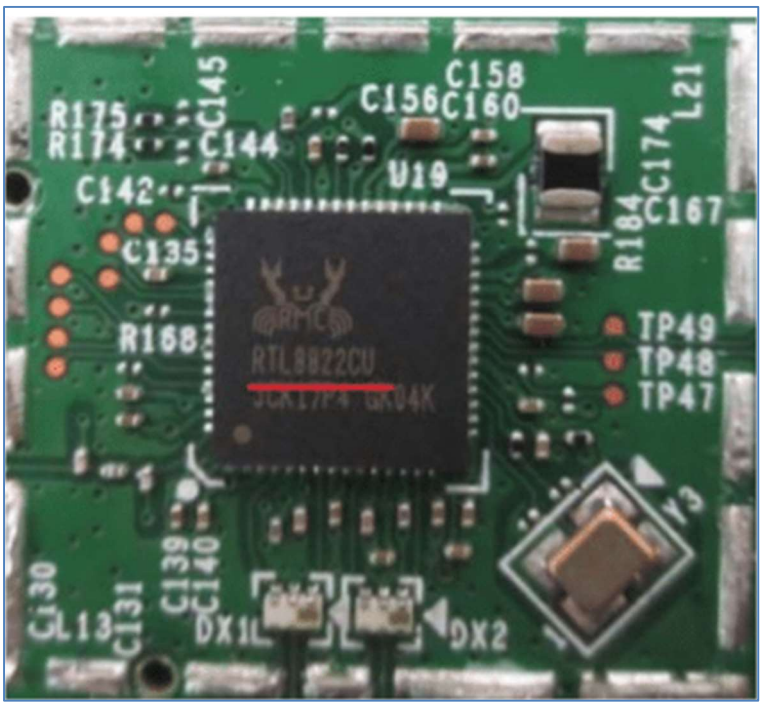




See e.g., https://www.microcontrollertips.com/wp-content/uploads/2020/12/PCB-top-and-bottom.jpg.

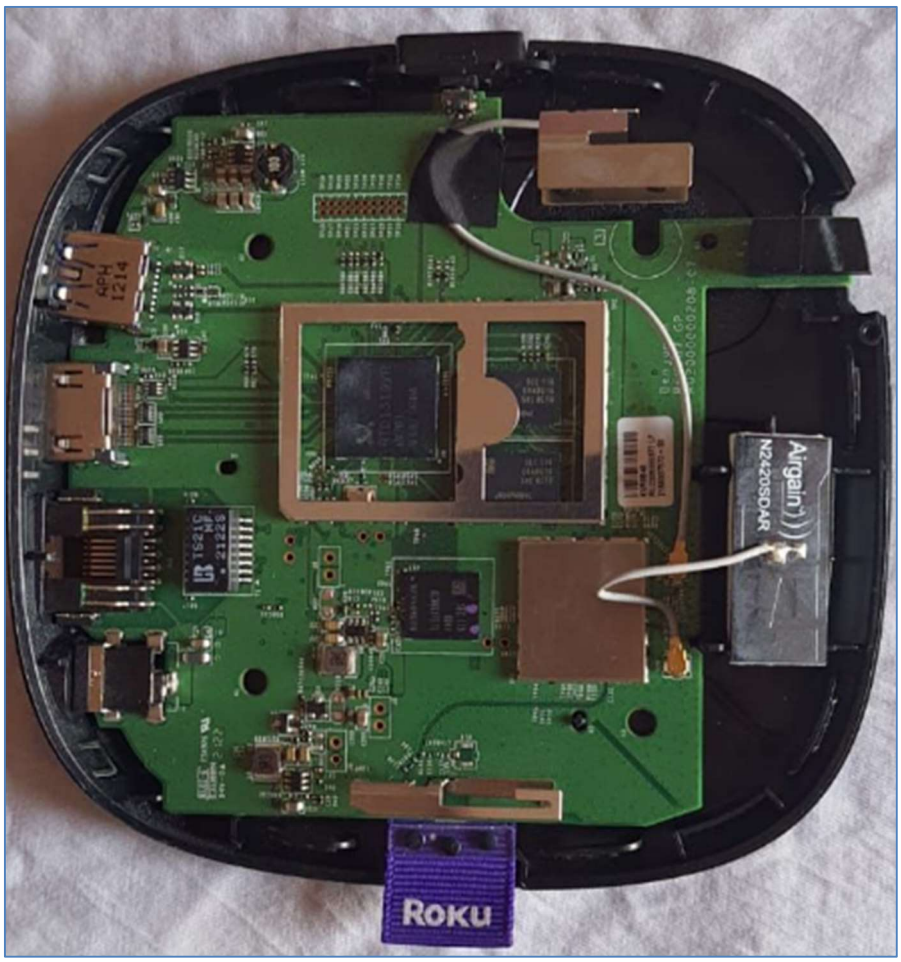


See e.g., images of teardown of the Roku Ultra.

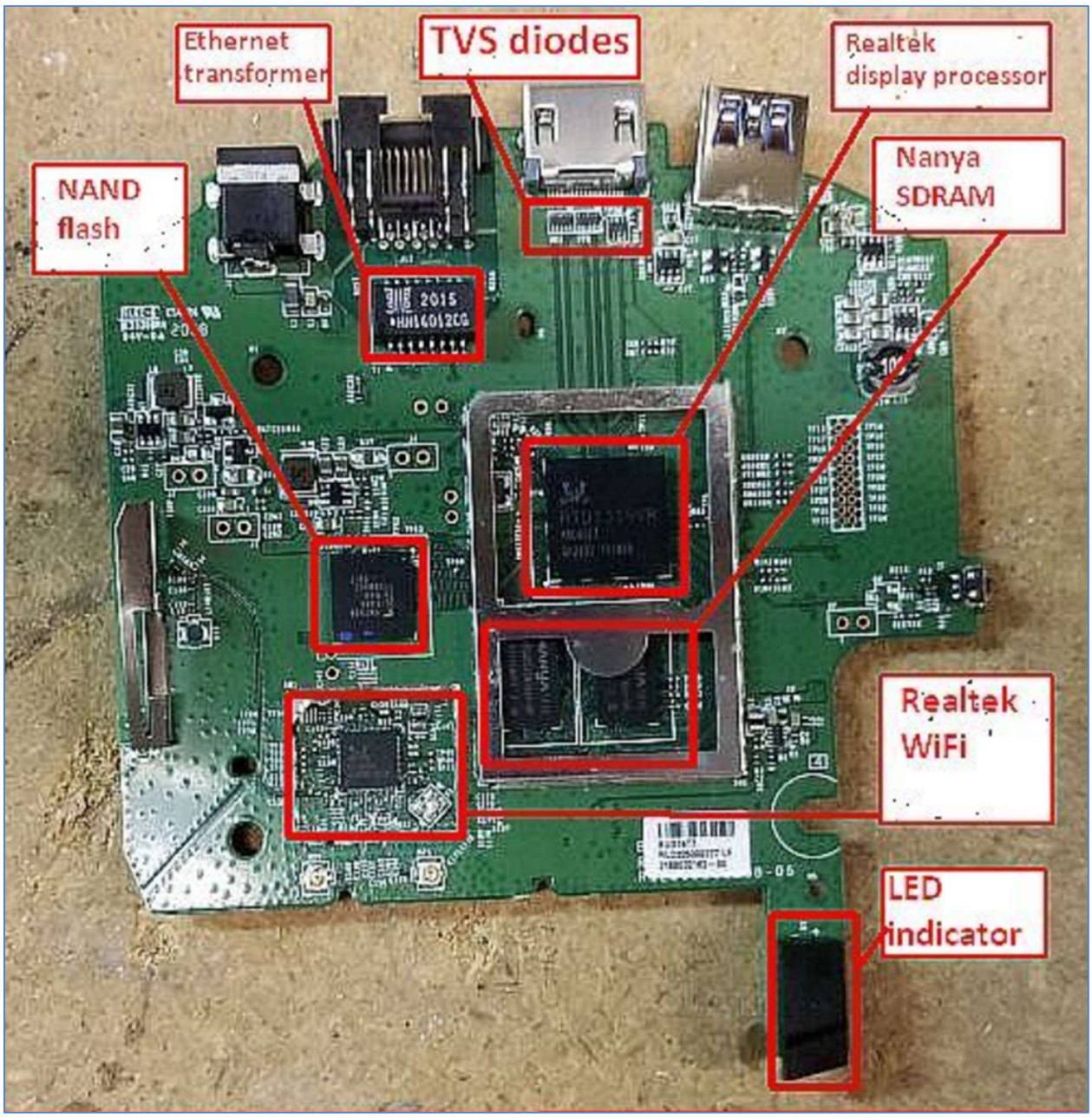


See e.g., from teardown of the Roku Ultra showing the display processor

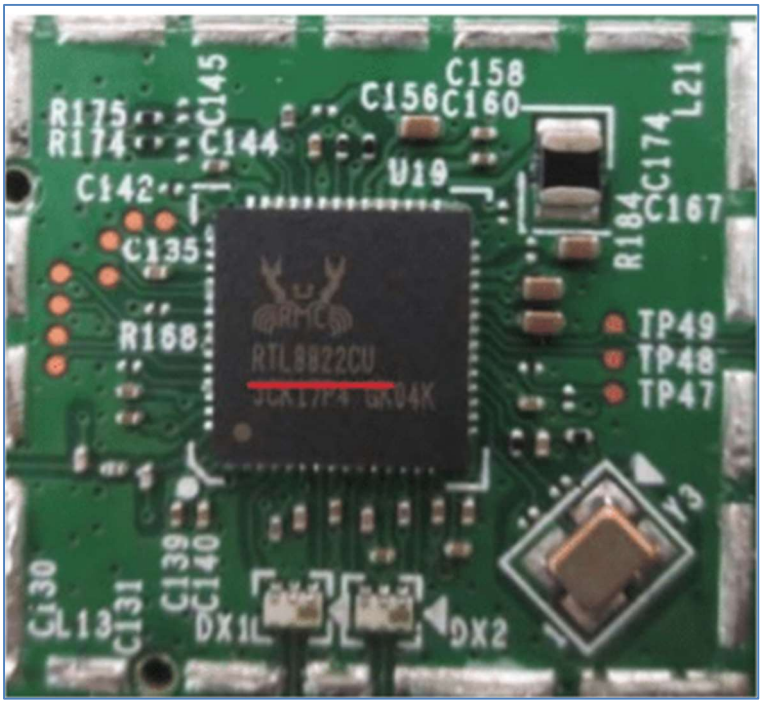
76. On information and belief, one or more components of the Roku system include a printed circuit board comprising a first point of reference potential and an input connector mounted to the input section of the printed circuit board, for coupling a signal from a signal source having a second point of reference potential to the printed circuit board.



See e.g., images of teardown of the Roku Ultra.



See e.g., https://www.microcontrollertips.com/wp-content/uploads/2020/12/PCB-top-and-bottom.jpg.



See e.g., from teardown of Roku Ultra showing the display processor.

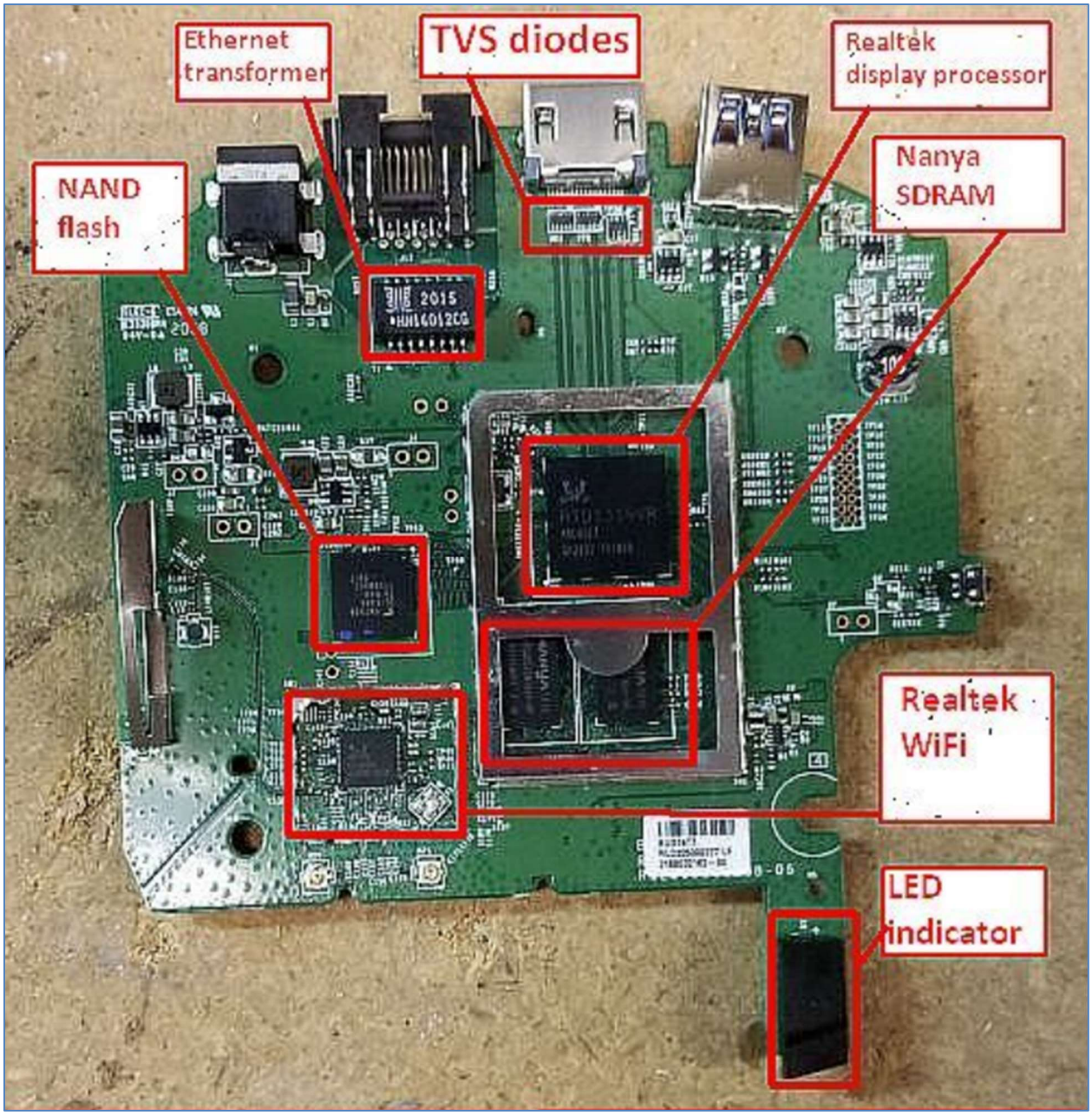
Detailed Parameters

Model No	6222D-UUC	Main Chipset	RTL8822CU
WIFI/BT Interface	USB2.0	Wireless Standards	IEEE802.11a/b/g/n/ac+BLE 5.0
Date Rates	867Mbps	Certification	RoHS,REACH
RF Antenna	External Antenn	OS Supported	Android /Linux/Windows
Operating Frequency	2.4/5.8GHz	Operating Voltage	3.3V
Dimensions	27x 18x2.03 mm	RF Chains	2T2R
Operating Temperature	0°C to 70°C	Storage Temperature	-40°C to 85°C

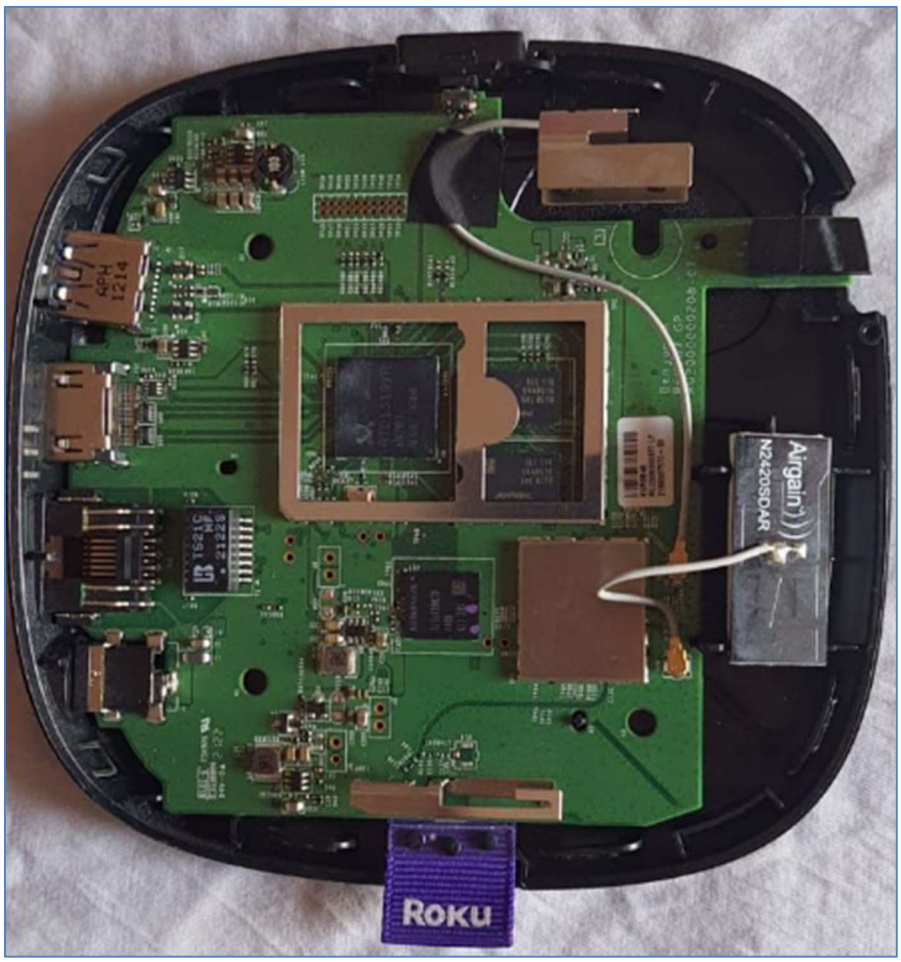
See e.g., https://www.alibaba.com/product-detail/5GHz-RTL8822CU-USB-To-WiFi-And_62562977566.html.

77. On information and belief, one or more components of the Roku system include a printed circuit board comprising a processing element mounted to the input

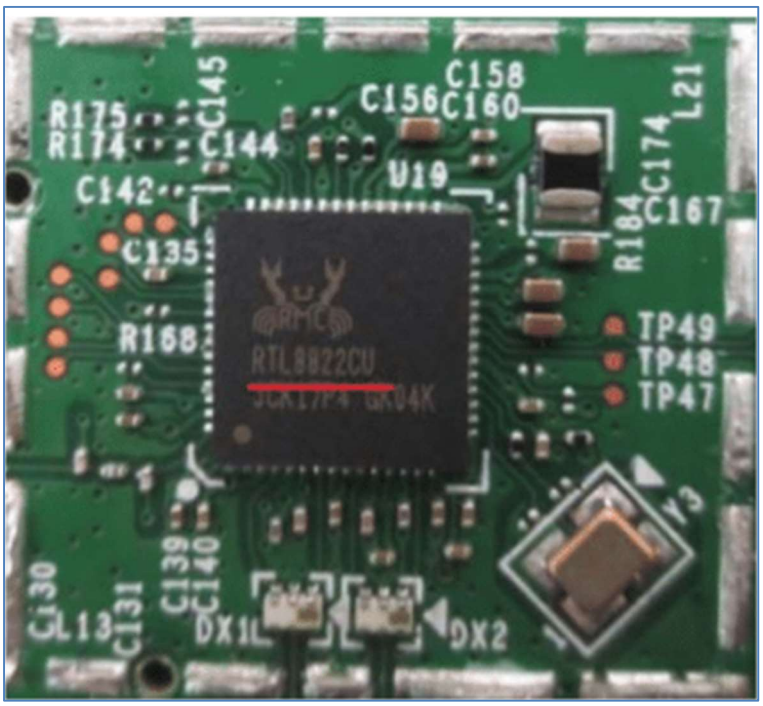
section of the printed circuit board, for processing a signal wherein the processing element is electrically coupled to the first point of reference potential.



See e.g., https://www.microcontrollertips.com/wp-content/uploads/2020/12/PCB-top-and-bottom.jpg.

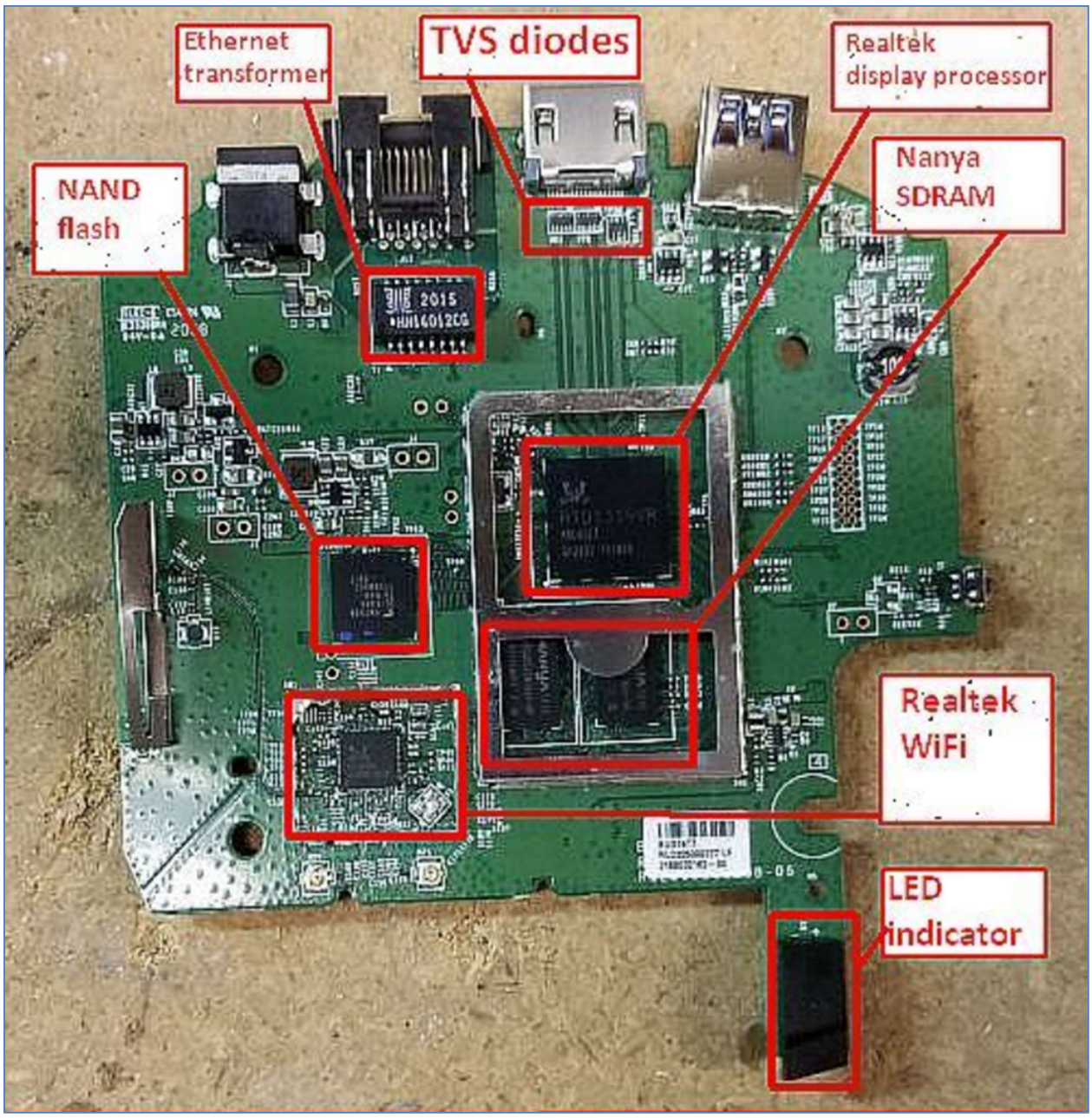


See e.g., images of teardown of the Roku Ultra.



See e.g., from teardown of Roku Ultra showing the display processor

78. On information and belief, one or more components of the Roku system include a printed circuit board comprising a shield wall, electrically coupled to said first point of reference potential, mounted between said input section and said output section of said printed circuit board and proximate to an output of said processing element, for electromagnetically isolating the input section of the printed circuit board from the output section of a printed circuit board, said shield wall having minimized apertures for passing said signals to the output section.



See e.g., https://www.microcontrollertips.com/wp-content/uploads/2020/12/PCB-top-and-bottom.jpg.

The first difference evident between this and previous generations of Roku devices is the heat sinking. Older Roku boxes had an aluminum piece screwed into the top half of the case. It provided a large heat-sink that cooled the main processor. The aluminum piece is gone from the new Ultra. Instead, designers used a smaller two-piece heat sink that fits on top of the metal shielding that protects the two SDRAM chips and the main display processor. Thermal grease connects the display processor to the heat sinking.

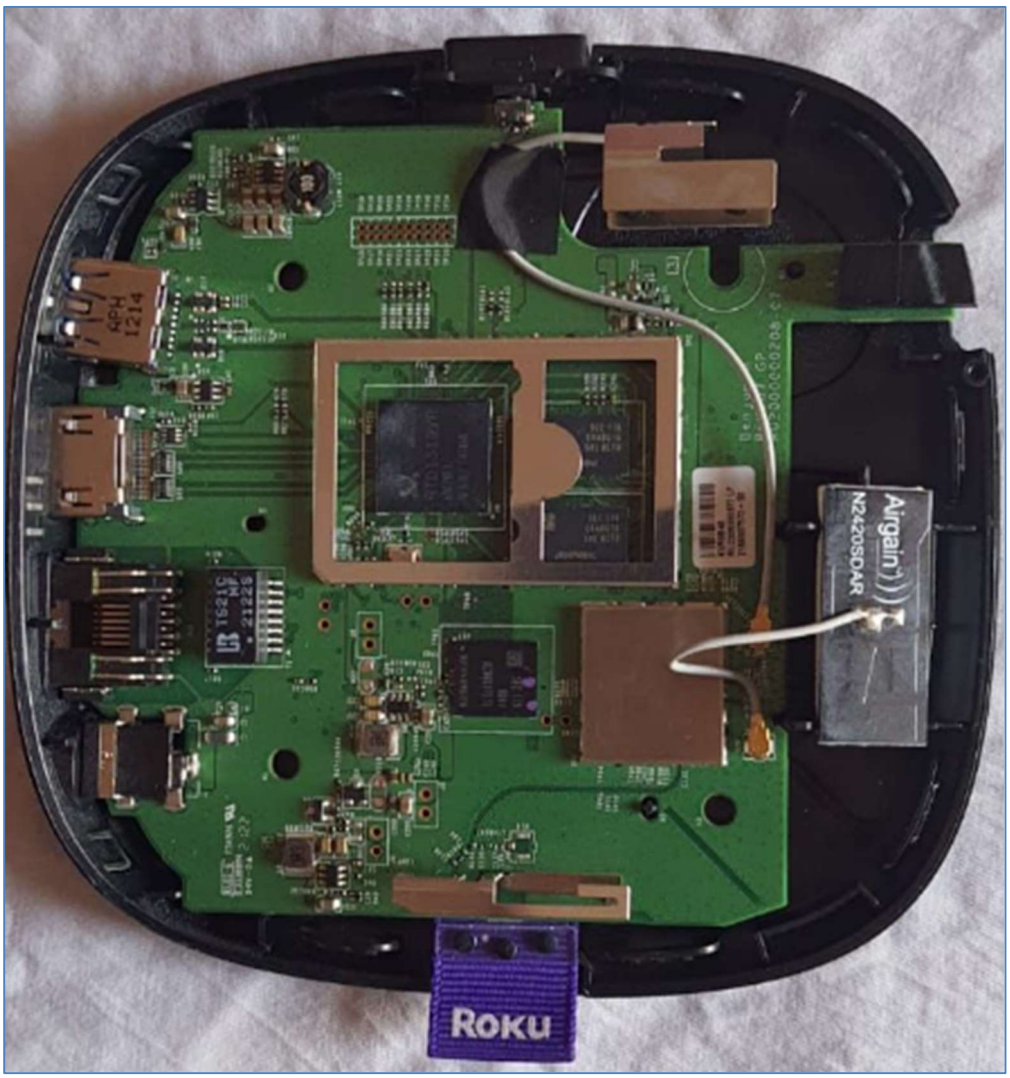
The other most evident difference in this edition of the Roku: The presence of three different antennas to handle the multiband WiFi and Bluetooth capabilities.

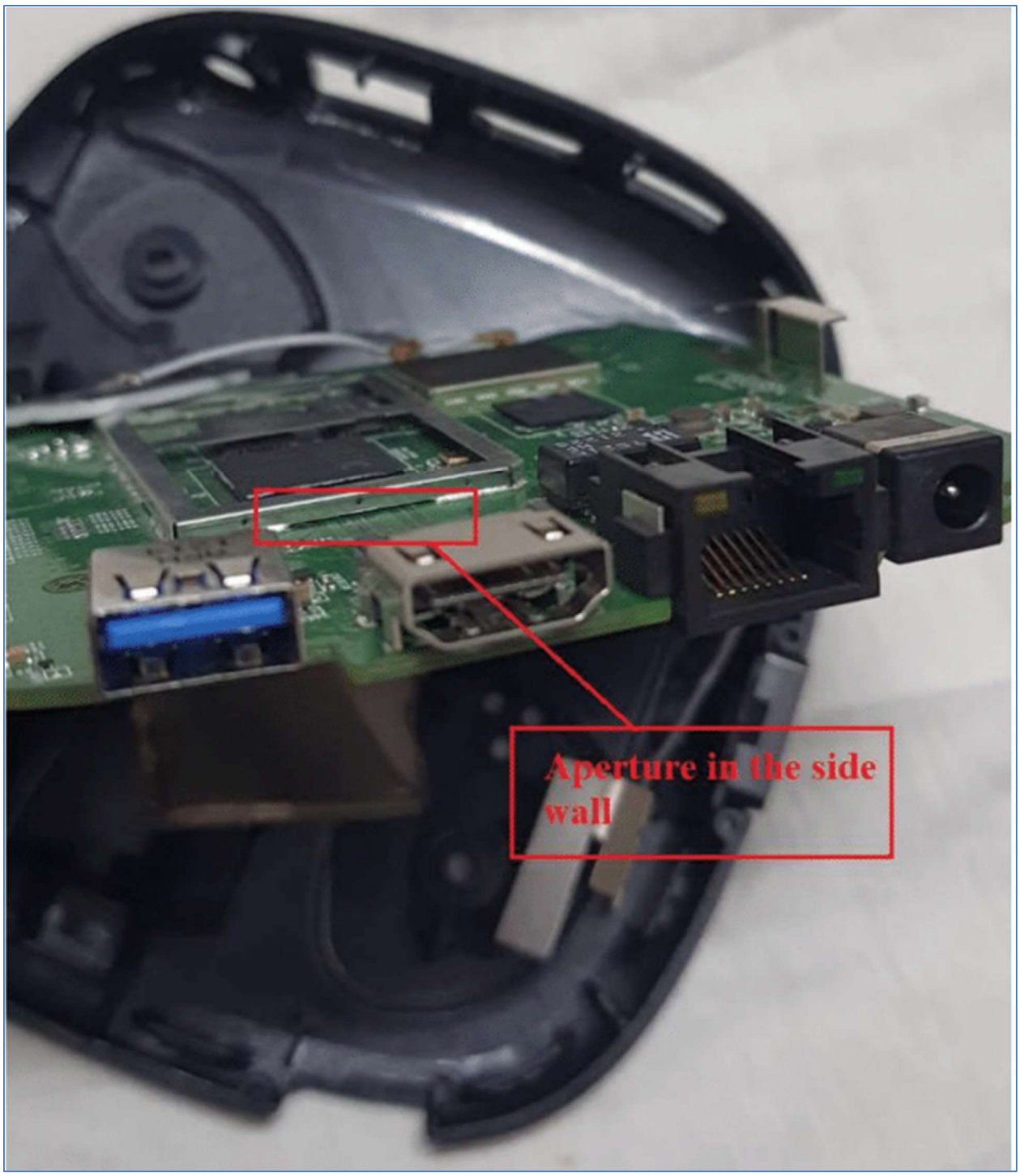
Most of the ICs on the Ultra come from Realtek in Taiwan. This is also a change in vendor from previous-generation

devices. The main display processor carries a part-number of RTD1319VR. This isn't a commercial Realtek device. The closest relative we could find is the RTD2662 series, a flat-panel LCD controller. It has some of the same features you'd expect to find in the Roku display processor, including support for an IR remote, HDMI processing, and an independent color processor. The Realtek display processor sits in a metal-shielded area together with two 8 Gb SDRAM chips from Nanya, a Taiwanese memory maker.

The two-piece heat sink in stages of disassembly. Click image to enlarge.

See e.g., https://www.microcontrollertips.com/teardown-the-new-roku-ultra/

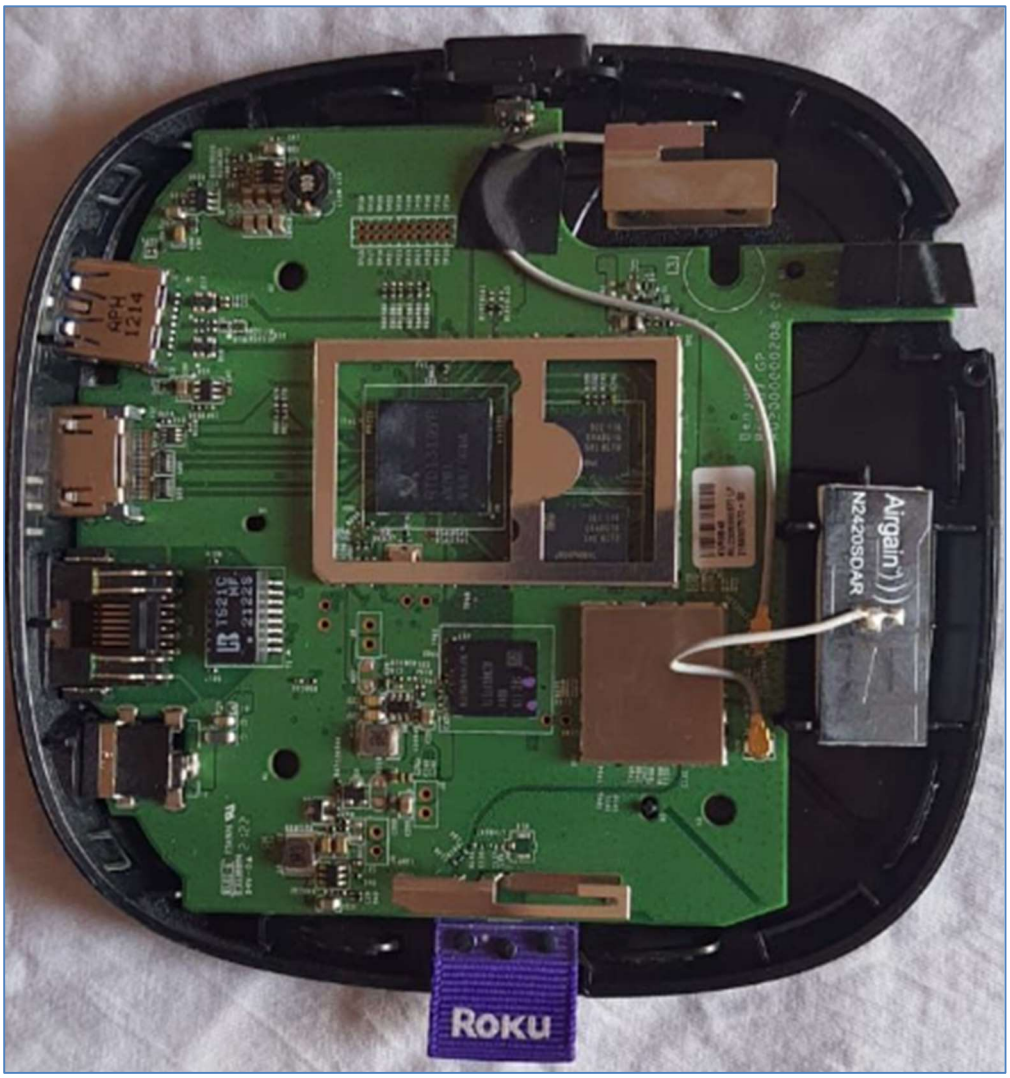

Airgain
N2420SOAR
ROKU



See e.g., images of teardown of the Roku Ultra.

79. On information and belief, one or more components of the Roku system include a printed circuit board comprising at least one output connector, mounted to

the output section of the printed circuit board, for outputting the processed signal to an electrical circuit included in a different printed circuit board having a third point of reference potential.



STEP 1: Connect to TV

Connect your streaming player to an HDMI port on the back of your TV with the included cable. For 4K streaming, you'll need to use an HDMI port that supports HDCP 2.2.

See e.g., https://image.roku.com/c3VwcG9ydC1B/Ultra-4800-QSG.pdf.

80. Roku's direct infringement has damaged AdaptFlow and caused it to suffer and continue to suffer irreparable harm and damages.

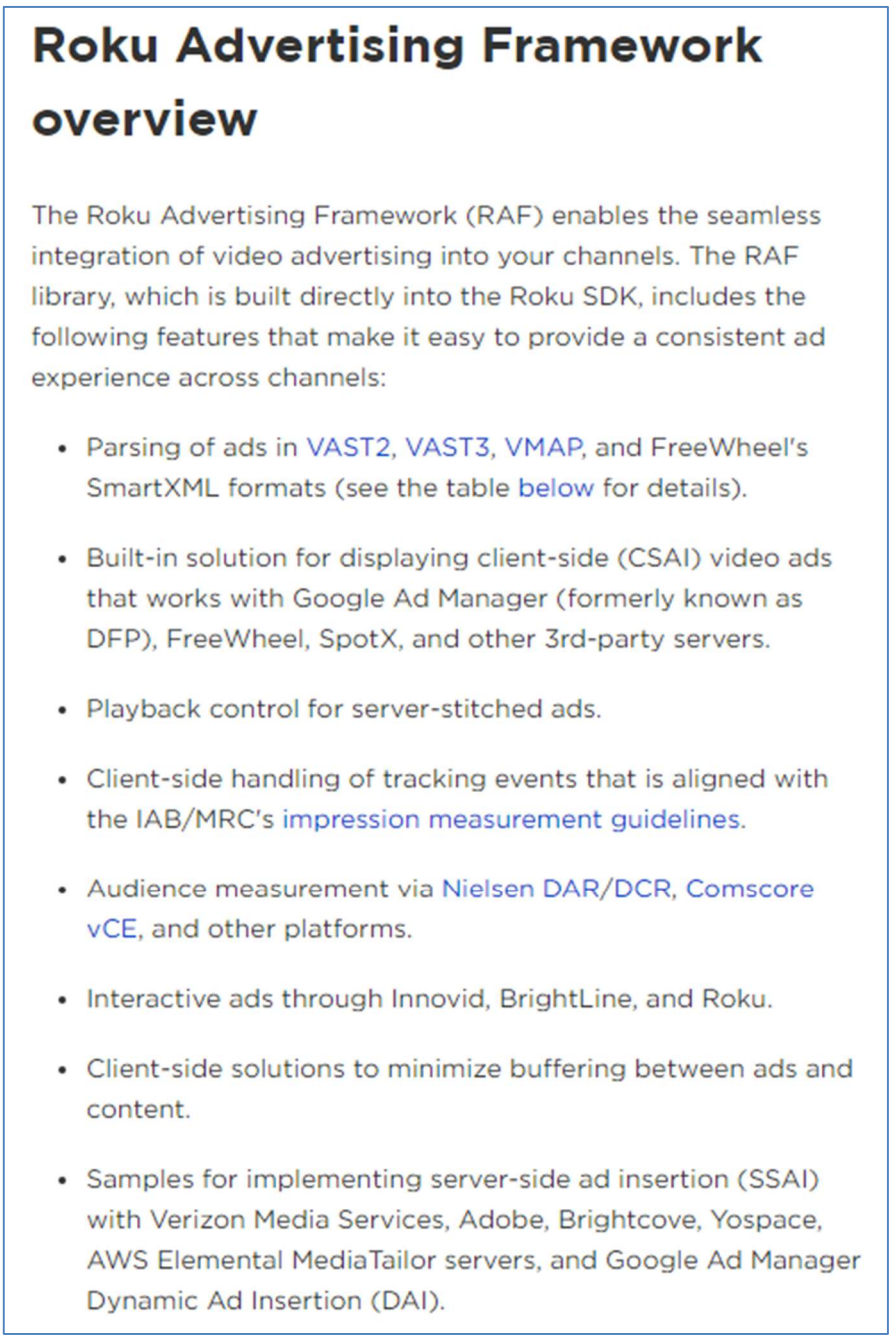
Count II – Infringement of United States Patent No. 7,206,494

81. AdaptFlow repeats, realleges, and incorporates by reference, as if fully set forth here, the allegations of the preceding paragraphs above.

82. On information and belief, Roku (or those acting on its behalf) makes, uses, sells, imports and/or offers to sell the Roku Products and Services; and makes, uses, sells, sells access to, imports, offers to sell and/or offers to sell access to the Roku

System in the United States that infringe (literally and/or under the doctrine of equivalents) at least claim 1 of the '494 patent.

83. On information and belief, one or more components of the Roku system employ and provide a method of identifying commercial message segments of a video signal.

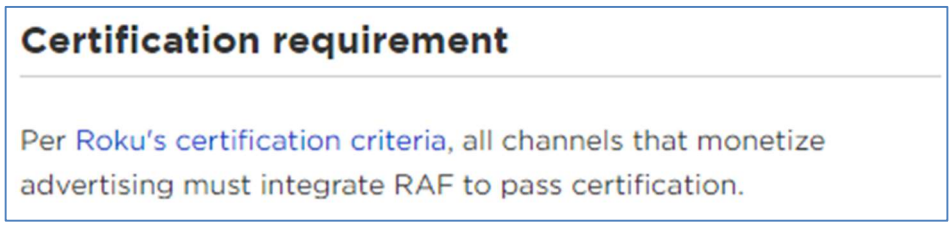


Roku Advertising Framework overview

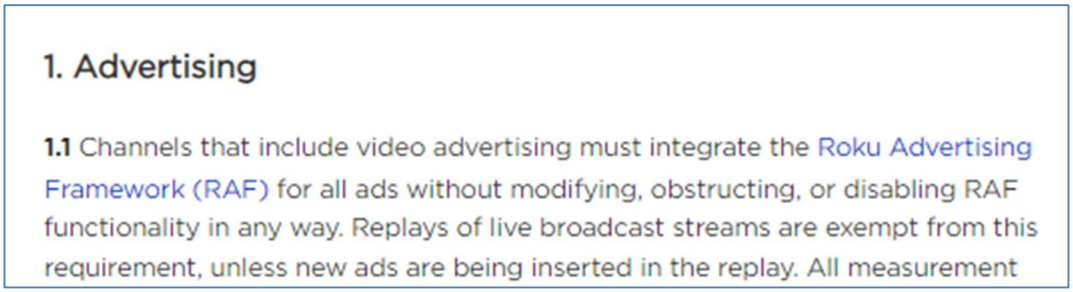
The Roku Advertising Framework (RAF) enables the seamless integration of video advertising into your channels. The RAF library, which is built directly into the Roku SDK, includes the following features that make it easy to provide a consistent ad experience across channels:

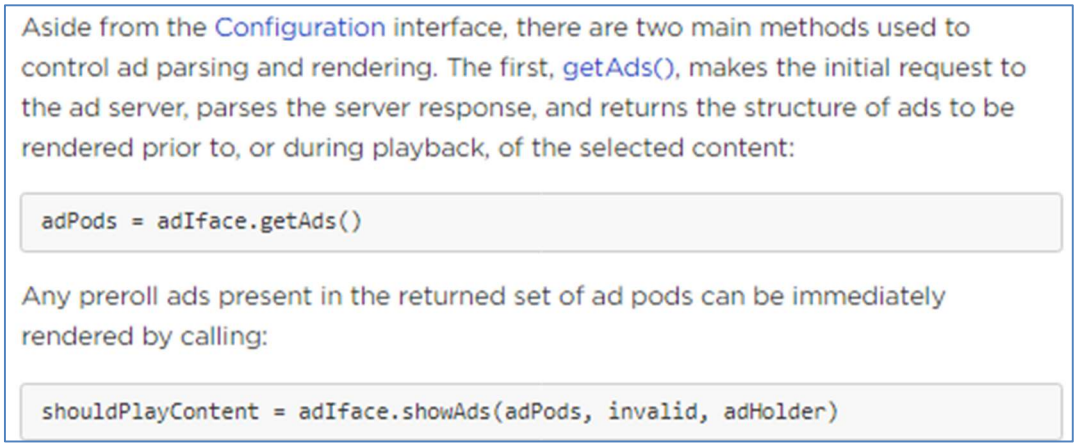
- Parsing of ads in VAST2, VAST3, VMAP, and FreeWheel's SmartXML formats (see the table below for details).
- Built-in solution for displaying client-side (CSAI) video ads that works with Google Ad Manager (formerly known as DFP), FreeWheel, SpotX, and other 3rd-party servers.
- Playback control for server-stitched ads.
- Client-side handling of tracking events that is aligned with the IAB/MRC's impression measurement guidelines.
- Audience measurement via Nielsen DAR/DCR, Comscore vCE, and other platforms.
- Interactive ads through Innovid, BrightLine, and Roku.
- Client-side solutions to minimize buffering between ads and content.
- Samples for implementing server-side ad insertion (SSAI) with Verizon Media Services, Adobe, Brightcove, Yospace, AWS Elemental MediaTailor servers, and Google Ad Manager Dynamic Ad Insertion (DAI).

See e.g., https://developer.roku.com/docs/developer-program/advertising/roku-advertising-framework.md.



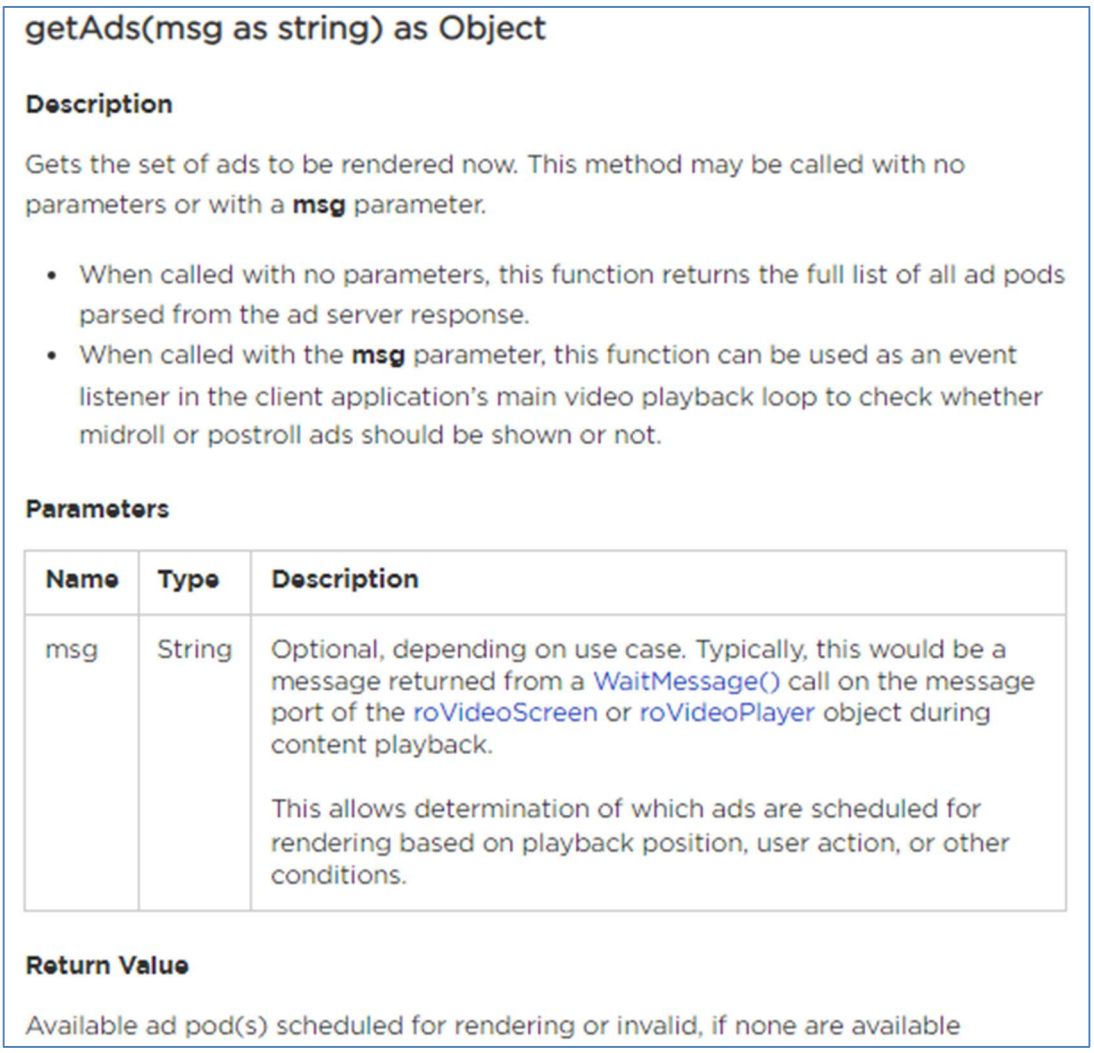
See e.g., https://developer.roku.com/docs/developer-program/advertising/roku-advertising-framework.md.





See e.g., https://developer.roku.com/docs/developer-program/advertising/integrating-roku-advertising-framework.md.

84. On information and belief, one or more components of the Roku system employ and provide a method for identifying commercial message segments of a video signal comprising the step of monitoring a digital bit stream comprising a video signal.



getAds(msg as string) as Object

Description

Gets the set of ads to be rendered now. This method may be called with no parameters or with a **msg** parameter.

- When called with no parameters, this function returns the full list of all ad pods parsed from the ad server response.
- When called with the **msg** parameter, this function can be used as an event listener in the client application's main video playback loop to check whether midroll or postroll ads should be shown or not.

Parameters

Name	Type	Description
msg	String	Optional, depending on use case. Typically, this would be a message returned from a WaitMessage() call on the message port of the roVideoScreen or roVideoPlayer object during content playback. This allows determination of which ads are scheduled for rendering based on playback position, user action, or other conditions.

Return Value

Available ad pod(s) scheduled for rendering or invalid, if none are available

See e.g., https://developer.roku.com/en-gb/docs/developer-program/advertising/raf-api.md.

Aside from the Configuration interface, there are two main methods used to control ad parsing and rendering. The first, getAds(), makes the initial request to the ad server, parses the server response, and returns the structure of ads to be rendered prior to, or during playback, of the selected content:

```
adPods = adIface.getAds()
```

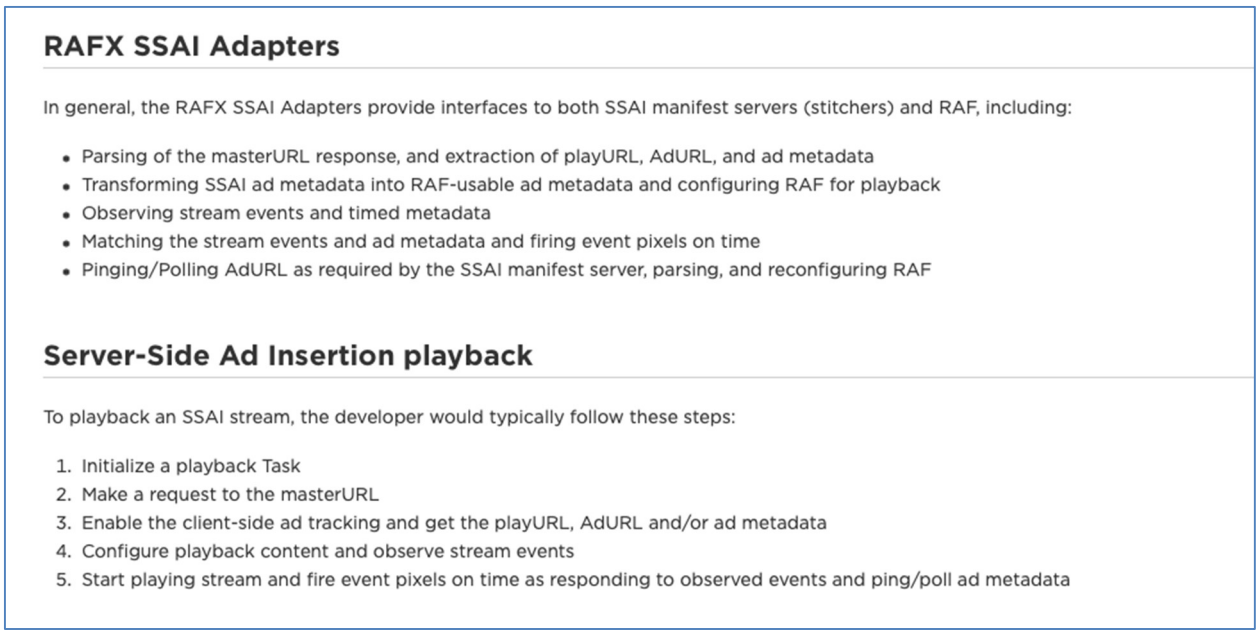
Any preroll ads present in the returned set of ad pods can be immediately rendered by calling:

```
shouldPlayContent = adIface.showAds(adPods, invalid, adHolder)
```

Complete ad rendering control example

```
shouldPlayContent = true
adBreakIndex = 0
while shouldPlayContent
  videoMsg = wait(0, contentVideoScreen.GetMessagePort())
  if videoMsg.isPlaybackPosition()
    curPos = videoMsg.GetIndex()
    nextPod = scheduledPods[adBreakIndex]
    if curPos > nextPod.renderTime and not nextPod.viewed
      contentVideoScreen.Close() ' stop playback of content
      shouldPlayContent = adIface.showAds(nextPod) ' render next ad pod
      adBreakIndex = adBreakIndex + 1
      if shouldPlayContent
        ' *** Insert client app's resume-playback code here
      end if
    end if
  end if
  ' *** Insert client app's video event handler code here
end while
```

See e.g., https://developer.roku.com/en-gb/docs/developer-program/advertising/integrating-roku-advertising-framework.md.



RAFX SSAI Adapters

In general, the RAFX SSAI Adapters provide interfaces to both SSAI manifest servers (stitchers) and RAF, including:

- Parsing of the masterURL response, and extraction of playURL, AdURL, and ad metadata
- Transforming SSAI ad metadata into RAF-usable ad metadata and configuring RAF for playback
- Observing stream events and timed metadata
- Matching the stream events and ad metadata and firing event pixels on time
- Pinging/Polling AdURL as required by the SSAI manifest server, parsing, and reconfiguring RAF

Server-Side Ad Insertion playback

To playback an SSAI stream, the developer would typically follow these steps:

1. Initialize a playback Task
2. Make a request to the masterURL
3. Enable the client-side ad tracking and get the playURL, AdURL and/or ad metadata
4. Configure playback content and observe stream events
5. Start playing stream and fire event pixels on time as responding to observed events and ping/poll ad metadata

See e.g., https://developer.roku.com/en-gb/docs/developer-program/advertising/ssai-adapters.md.

85. On information and belief, one or more components of the Roku system employ and provide a method for identifying commercial message segments of a video signal comprising detecting a change in an informational parameter of said video signal exclusive of audio-visual content.

getAds(msg as string) as Object

Description

Gets the set of ads to be rendered now. This method may be called with no parameters or with a **msg** parameter.

- When called with no parameters, this function returns the full list of all ad pods parsed from the ad server response.
- When called with the **msg** parameter, this function can be used as an event listener in the client application's main video playback loop to check whether midroll or postroll ads should be shown or not.

Parameters

Name	Type	Description
msg	String	Optional, depending on use case. Typically, this would be a message returned from a WaitMessage() call on the message port of the roVideoScreen or roVideoPlayer object during content playback. This allows determination of which ads are scheduled for rendering based on playback position, user action, or other conditions.

Return Value

Available ad pod(s) scheduled for rendering or invalid, if none are available

See e.g., https://developer.roku.com/en-gb/docs/developer-program/advertising/raf-api.md#client-ad-insertion.

adBreaks	string	Required if monetizing content	One or more time codes. Represents a time duration from the beginning of the video where an ad shows up. Conforms to the format: {hh}:{mm}:{ss} and in the form of an SCTE-35 marker. See each content type for its ad policy.

See e.g., https://developer.roku.com/en-gb/docs/specs/direct-publisher-feed-specs/json-dp-spec.md#content.

setAdBreaks(contentLength as Integer, adBreakTimes as Integer)

Description

Configures content playback parameters, which can be used for scheduling relative-positioned ad breaks in VMAP ad service responses.

- If your application uses VMAP ad URLs and they are configured to use "nn%" timeOffset values, then you must specify the contentLength prior to calling getAds().
- If VMAP is configured to use "#mm" timeOffset values, you must first specify a set of ad break times.
- Calling with empty parameters will reset these to invalid values.

The content length can also be set independently via setContentLength() if ad break times are not required.

Parameters

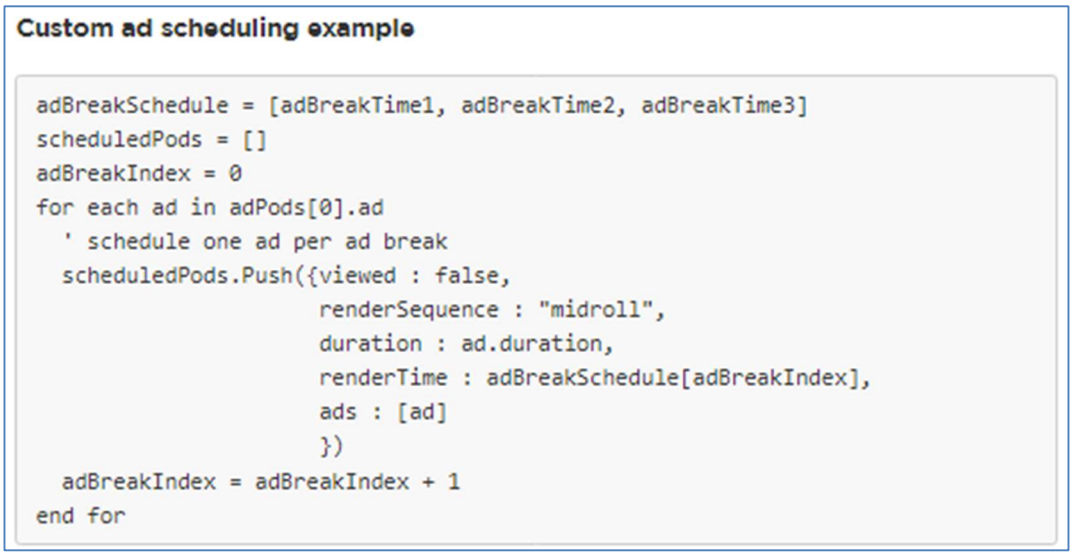
Argument	Type	Description
contentLength	Integer	Total length of video content (in seconds).
adBreakTimes	Integer	Array of suggested offsets into content playback to insert ad breaks (in seconds).

See e.g., https://developer.roku.com/en-gb/docs/developer-program/advertising/raf-api.md#client-ad-insertion.

Calling getAds() in a while loop

```
while shouldPlayContent
  videoMsg = wait(0, contentVideoScreen.GetMessagePort())
  adPods = adIface.getAds(videoMsg)
  if adPods <> invalid and adPods.Count() > 0
    contentVideoScreen.Close() ' stop playback of content
    shouldPlayContent = adIface.showAds(adPods) ' render current ad pod
    if shouldPlayContent
      ' *** Insert client app's resume-playback code here
    end if
  end if
  ' *** Insert client app's video event handler code here
end while
```

See e.g., https://developer.roku.com/en-gb/docs/developer-program/advertising/integrating-roku-advertising-framework.md#custom-ad-parsing-and-rendering.



Custom ad scheduling example

```
adBreakSchedule = [adBreakTime1, adBreakTime2, adBreakTime3]
scheduledPods = []
adBreakIndex = 0
for each ad in adPods[0].ad
  ' schedule one ad per ad break
  scheduledPods.Push({viewed : false,
                      renderSequence : "midroll",
                      duration : ad.duration,
                      renderTime : adBreakSchedule[adBreakIndex],
                      ads : [ad]
                      })
  adBreakIndex = adBreakIndex + 1
end for
```

See e.g., https://developer.roku.com/en-gb/docs/developer-program/advertising/integrating-roku-advertising-framework.md#custom-ad-parsing-and-rendering.

Complete ad rendering control example

```
shouldPlayContent = true
adBreakIndex = 0
while shouldPlayContent
  videoMsg = wait(0, contentVideoScreen.GetMessagePort())
  if videoMsg.isPlaybackPosition()
    curPos = videoMsg.GetIndex()
    nextPod = scheduledPods[adBreakIndex]
    if curPos > nextPod.renderTime and not nextPod.viewed
      contentVideoScreen.Close() ' stop playback of content
      shouldPlayContent = adIface.showAds(nextPod) ' render next ad pod
      adBreakIndex = adBreakIndex + 1
      if shouldPlayContent
        ' *** Insert client app's resume-playback code here
      end if
    end if
  end if
  ' *** Insert client app's video event handler code here
end while
```

See e.g., https://developer.roku.com/en-gb/docs/developer-program/advertising/integrating-roku-advertising-framework.md#custom-ad-parsing-and-rendering.

86. On information and belief, one or more components of the Roku system employ and provide a method for identifying commercial message segments of a video signal comprising selectively generating a commercial event notification responsive to the detecting step, wherein said detection step additionally comprises detecting the presence of a splice table in the digital bit stream.

Calling getAds() in a while loop

```
while shouldPlayContent
  videoMsg = wait(0, contentVideoScreen.GetMessagePort())
  adPods = adIface.getAds(videoMsg)
  if adPods <> invalid and adPods.Count() > 0
    contentVideoScreen.Close() ' stop playback of content
    shouldPlayContent = adIface.showAds(adPods) ' render current ad pod
    if shouldPlayContent
      ' *** Insert client app's resume-playback code here
    end if
  end if
  ' *** Insert client app's video event handler code here
end while
```

See e.g., https://developer.roku.com/en-gb/docs/developer-program/advertising/integrating-roku-advertising-framework.md#custom-ad-parsing-and-rendering.

Complete ad rendering control example

```
shouldPlayContent = true
adBreakIndex = 0
while shouldPlayContent
  videoMsg = wait(0, contentVideoScreen.GetMessagePort())
  if videoMsg.isPlaybackPosition()
    curPos = videoMsg.GetIndex()
    nextPod = scheduledPods[adBreakIndex]
    if curPos > nextPod.renderTime and not nextPod.viewed
      contentVideoScreen.Close() ' stop playback of content
      shouldPlayContent = adIface.showAds(nextPod) ' render next ad pod
      adBreakIndex = adBreakIndex + 1
      if shouldPlayContent
        ' *** Insert client app's resume-playback code here
      end if
    end if
  end if
  ' *** Insert client app's video event handler code here
end while
```

See e.g., https://developer.roku.com/en-gb/docs/developer-program/advertising/integrating-roku-advertising-framework.md#custom-ad-parsing-and-rendering.

Custom ad scheduling example

```
adBreakSchedule = [adBreakTime1, adBreakTime2, adBreakTime3]
scheduledPods = []
adBreakIndex = 0
for each ad in adPods[0].ad
  ' schedule one ad per ad break
  scheduledPods.Push({viewed : false,
                      renderSequence : "midroll",
                      duration : ad.duration,
                      renderTime : adBreakSchedule[adBreakIndex],
                      ads : [ad]
                      })
  adBreakIndex = adBreakIndex + 1
end for
```

See e.g., https://developer.roku.com/en-gb/docs/developer-program/advertising/integrating-roku-advertising-framework.md#custom-ad-parsing-and-rendering.

87. On information and belief, one or more components of the Roku system employ and provide a method for identifying commercial message segments of a video signal comprising selectively generating a commercial event notification responsive to the detecting step, wherein said detection step additionally comprises comparing a timing of the commercial event notification indicating the occurrence of a commercial message, to commercial message insertion data contained in the splice table and verifying the presence of a commercial message based on the comparing step.

Complete ad rendering control example

```
shouldPlayContent = true
adBreakIndex = 0
while shouldPlayContent
  videoMsg = wait(0, contentVideoScreen.GetMessagePort())
  if videoMsg.isPlaybackPosition()
    curPos = videoMsg.GetIndex()
    nextPod = scheduledPods[adBreakIndex]
    if curPos > nextPod.renderTime and not nextPod.viewed
      contentVideoScreen.Close() ' stop playback of content
      shouldPlayContent = adIface.showAds(nextPod) ' render next ad pod
      adBreakIndex = adBreakIndex + 1
      if shouldPlayContent
        ' *** Insert client app's resume-playback code here
      end if
    end if
  end if
  ' *** Insert client app's video event handler code here
end while
```

See e.g., https://developer.roku.com/en-gb/docs/developer-program/advertising/integrating-roku-advertising-framework.md#custom-ad-parsing-and-rendering.

showAds(ads as Object, ctx as Object, view as Object) as Boolean

Description

Renders any ads scheduled for display.

When this method is called with an array of ad pods (for example, using the value returned from the initial call to the getAds() method), this is interpreted to mean that any preroll ad pod present should be rendered.

Client applications should always check the return value. If it is false, an application should exit content playback and return to the content selection screen. Typically, this occurs when the user presses the "Back" button during ad playback.

Parameters

Argument	Type	Required?	Description
ads	array of ad pods	required	Ads to be rendered. Can represent either a single pod of ads or an array of ad pods.
ctx	associative array	optional	An associative array that allows client code to provide new offset and total to ad counter to support use cases involving interleaving RAF rendering with custom rendering within a single pod of ads. When used, it should be in the form of: `{ start: Integer, total: Integer }` For example, `{ start: 1, total: 4 }` would display as: "Ad 1 of 4" in the top left corner during ad playback.
view	renderable node	required (for SceneGraph applications)	Parameter representing a renderable node to which the ad UI can be parented. The **view** parameter allows SceneGraph rendering of ads into an app that uses SceneGraph for content rendering. • For server-stitched use case, this should be the Video node of the content player. • For non-stitched use cases, this can be any renderable node in the scene whose lifetime is guaranteed during the duration of ad rendering. Render any ads scheduled for display. The dimensions of the view object will be used to position RAF's UI elements, so it must be properly sized. Having dimensions larger than the current video playback resolution can place RAF UI elements such as the progress bar off screen.

Return Value

A flag indicating whether the ad pod was rendered to completion. This will be false if the user exited before render completion.

See e.g., https://developer.roku.com/en-gb/docs/developer-program/advertising/raf-api.md.

88. Roku's direct infringement has damaged AdaptFlow and caused it to suffer and continue to suffer irreparable harm and damages.

Count III – Infringement of United States Patent No. 7,340,528

89. AdaptFlow repeats, realleges, and incorporates by reference, as if fully set forth here, the allegations of the preceding paragraphs above.

90. On information and belief, Roku (or those acting on its behalf) makes, uses, sells, imports and/or offers to sell the Roku Products and Services; and makes, uses, sells, sells access to, imports, offers to sell and/or offers to sell access to the Roku System in the United States that infringe (literally and/or under the doctrine of equivalents) at least claim 1 of the '528 patent.

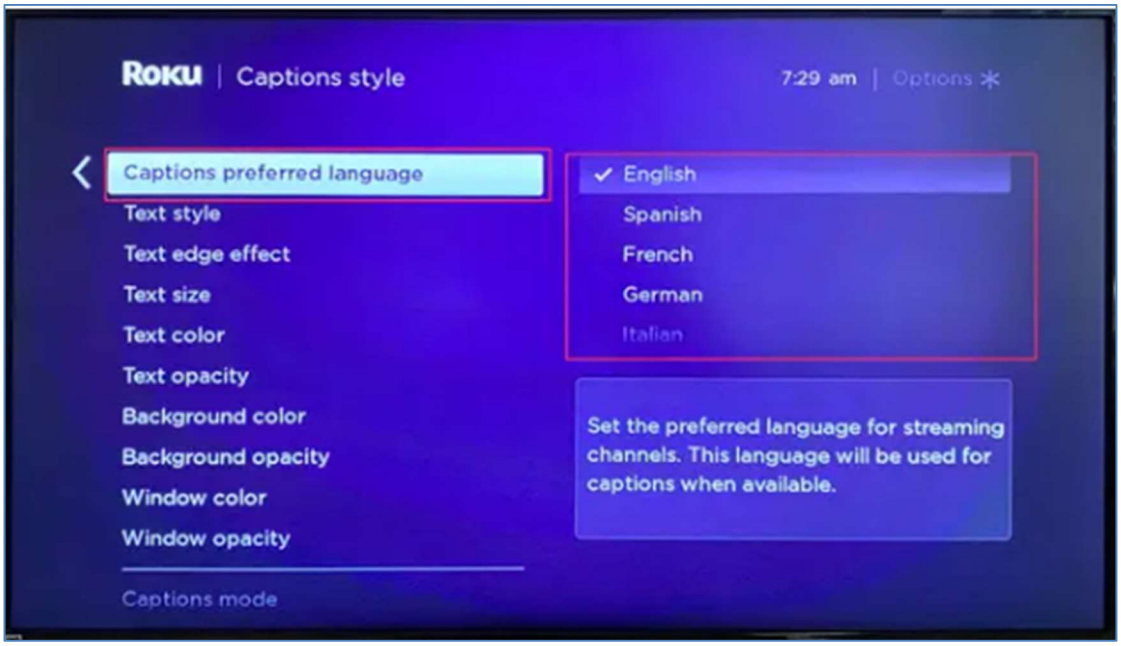
91. On information and belief, one or more components of the Roku system employ and provide a method of preparing a received content element for presentation.

Closed caption support summary

Below is a summary of the closed caption formats supported by the various video streaming technologies on Roku devices.

	SMPTE-TT	EIA-608	WebVTT
MP4 VOD	Yes (external file only)	Yes (in stream only for all manifests containing AVC streams)	No
HLS VOD	Yes (external file only)	Yes (in stream only for all manifests containing AVC streams)	Yes (in a separate stream described in the manifest (playlist for hls)
HLS Live	No	Yes (in stream only for all manifests containing AVC streams)	Yes (in a separate stream described in the manifest (playlist for hls)
Smooth VOD	Yes (in stream or external file)	Yes (in stream only for all manifests containing AVC streams)	No
Smooth Live	Yes (in stream only)	Yes (in stream only for all manifests containing AVC streams)	No
DASH VOD	Yes (external file only)	Yes (in stream only for all manifests containing AVC streams)	Yes
DASH Live	Yes (in stream or external file)	Yes (in stream only for all manifests containing AVC streams)	Yes

See e.g., https://developer.roku.com/en-gb/docs/developer-program/media-playback/closed-caption.md.



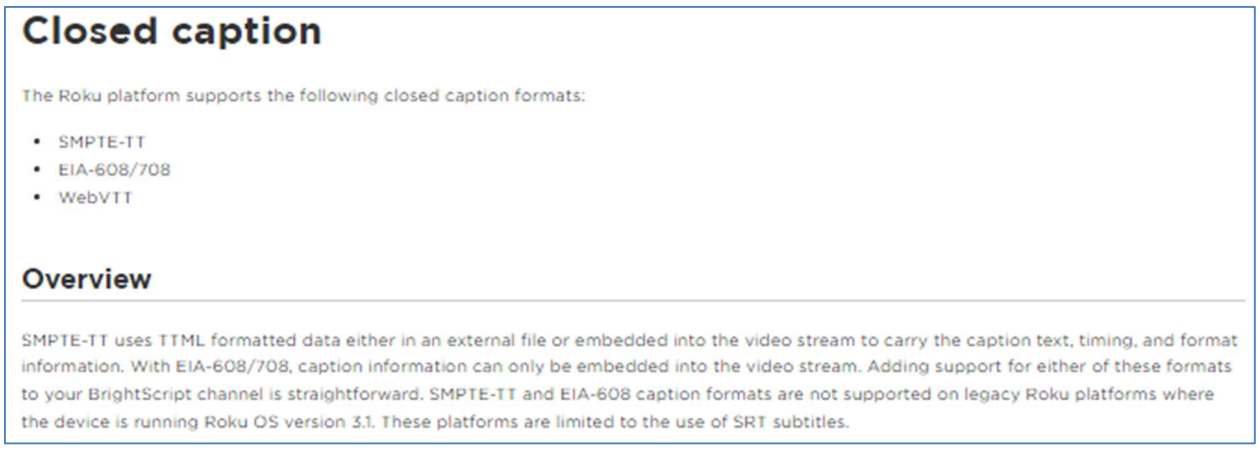
See e.g., https://www.businessinsider.com/how-to-change-language-on-roku.

The following is a list of recommendations and items to be aware of:

- Starting from Roku OS 8, it is no longer necessary for a channel to partake in the closed caption track selection, apart from adding any available tracks to the list of available tracks. the Roku OS now selects a closed caption track based on the preferred caption language selection in the system preferences. When the selected language is not available, it defaults to the system's UI language.
- The global **closedCaptionMode** method of the Video node object is how you turn on and off closed captioning of the current playing video. The global settings can be read and set in the rodeviceinfo object. These affect the same system setting. Whenever the user switches on/off closed caption, it is expected that the global setting will be adjusted accordingly. Therefore setting the global setting every time you adjust a local setting is required.
- The **audio track** and the **subtitle track** (for Multilanguage subtitles) can be set using the **VideoNode.audioTrack** and **VideoNode.subtitleTrack** respectively. The available tracks can be found with **VideoNode.availableAudioTracks** and **VideoNode.availableSubtitleTracks**. Another useful item is **rodeviceInfo.GetCurrentLocale**.
- If you are using the **roVideoScreen** or **roVideoPlayer**, you should be rewriting your application in **SceneGraph** as the older technologies are being dropped from the Roku OS.

See e.g., https://developer.roku.com/en-gb/docs/developer-program/media-playback/closed-caption.md.

92. On information and belief, one or more components of the Roku system employ and provide a method of preparing a received content element for presentation comprising the step of receiving data, including a content element.



Closed caption

The Roku platform supports the following closed caption formats:

- SMPTE-TT
- EIA-608/708
- WebVTT

Overview

SMPTE-TT uses TTML formatted data either in an external file or embedded into the video stream to carry the caption text, timing, and format information. With EIA-608/708, caption information can only be embedded into the video stream. Adding support for either of these formats to your BrightScript channel is straightforward. SMPTE-TT and EIA-608 caption formats are not supported on legacy Roku platforms where the device is running Roku OS version 3.1. These platforms are limited to the use of SRT subtitles.

See e.g., https://developer.roku.com/en-gb/docs/developer-program/media-playback/closed-caption.md.

Closed caption support summary

Below is a summary of the closed caption formats supported by the various video streaming technologies on Roku devices.

	SMPTE-TT	EIA-608	WebVTT
MP4 VOD	Yes (external file only)	Yes (in stream only for all manifests containing AVC streams)	No
HLS VOD	Yes (external file only)	Yes (in stream only for all manifests containing AVC streams)	Yes (in a separate stream described in the manifest (playlist for hls)
HLS Live	No	Yes (in stream only for all manifests containing AVC streams)	Yes (in a separate stream described in the manifest (playlist for hls)
Smooth VOD	Yes (in stream or external file)	Yes (in stream only for all manifests containing AVC streams)	No
Smooth Live	Yes (in stream only)	Yes (in stream only for all manifests containing AVC streams)	No
DASH VOD	Yes (external file only)	Yes (in stream only for all manifests containing AVC streams)	Yes
DASH Live	Yes (in stream or external file)	Yes (in stream only for all manifests containing AVC streams)	Yes

See e.g., https://developer.roku.com/en-gb/docs/developer-program/media-playback/closed-caption.md.

93. On information and belief, one or more components of the Roku system employ and provide a method of preparing a received content element for presentation comprising the step of reviewing the received data to identify a reference associated with the content element.

<table>
<tr><td>subtitleTrack</td><td>string</td><td></td><td>READ_WRITE</td><td>The identifier of the selected subtitle track. Subtitles may or may not be visible on the screen, depending upon the user's caption mode setting.

Reading this field will return the identifier of the subtitle track selected by the user. Writing this the field will change the track.

See also: globalCaptionMode</td></tr>
<tr><td>currentSubtitleTrack</td><td>string</td><td></td><td>READ_ONLY</td><td>The identifier of the selected subtitle track. Subtitles may or may not be visible on the screen, depending upon the user's caption mode setting.

Reading this field will return the identifier of the subtitle track that is playing. When the user has not selected a track, the Roku media player will select a track based on the preferred caption language system setting.</td></tr>
<tr><td>availableSubtitleTracks</td><td>array of associative arrays</td><td>[ ] empty array</td><td>READ_ONLY</td><td>The list of subtitle tracks available in the video stream. The array is initially populated with the tracks specified in the Content Meta-Data, and additional tracks are added if they are detected by the digital video player. Each associative array has the following entries:

<table>
<tr><th>Key</th><th>Type</th><th>Value</th></tr>
<tr><td>Description</td><td>string</td><td>Descriptive name of the subtitle track</td></tr>
<tr><td>Language</td><td>string</td><td>ISO 639-2 three-letter language code</td></tr>
<tr><td>TrackName</td><td>string</td><td>The track identifier. The value of this field may be used to select the subtitle track.</td></tr>
</table>
</td></tr>
</table>

<table>
<tr>
<td>captionStyle</td>
<td>associative array</td>
<td>system default</td>
<td>READ_WRITE</td>
<td>Available since Roku OS 8.

Allows channels to style closed captions. For any keys that are absent from the associative array, or for unexpected values, the Default value is assumed for that property. Following are the possible key names and values for this field:

<table>
<tr><th>Property</th><th>Possible Values</th></tr>
<tr><td>Text/Font</td><td>Default
Serif Fixed Width
Serif Proportional
Sans Serif Fixed Width
Sans Serif Proportional
Casual
Cursive
Small Caps</td></tr>
<tr><td>Text/Effect</td><td>Default
None
Raised
Depressed
Uniform
Drop shadow (left)
Drop shadow (right)</td></tr>
<tr><td>Text/Size</td><td>Default
Large
Medium
Small</td></tr>
<tr><td>Text/Color</td><td>Default
White
Black
Red
Green
Blue
Yellow
Magenta
Cyan</td></tr>
<tr><td>Text/Opacity</td><td>Default
25%
50%
75%
100%</td></tr>
<tr><td>Background/Color</td><td>Default
White
Black
Red
Green
Blue
Yellow
Magenta
Cyan</td></tr>
<tr><td>Background/Opacity</td><td>Default
Off
25%
50%
75%
100%</td></tr>
<tr><td>Window/Color</td><td>Default
White
Black
Red
Green
Blue
Yellow
Magenta
Cyan</td></tr>
<tr><td>Window/Opacity</td><td>Default
Off
25%
50%
75%
100%</td></tr>
</table>
</td>
</tr>
</table>

See e.g., https://developer.roku.com/en-gb/docs/references/scenegraph/media-playback-nodes/video.md#closed-caption-fields.

94. On information and belief, one or more components of the Roku system employ and provide a method of preparing a received content element for presentation comprising the step of using the reference to determine the presentation configuration for the content element.

<table>
<tr><td>subtitleTrack</td><td>string</td><td></td><td>READ_WRITE</td><td>The identifier of the selected subtitle track. Subtitles may or may not be visible on the screen, depending upon the user's caption mode setting.

Reading this field will return the identifier of the subtitle track selected by the user. Writing this the field will change the track.

See also: globalCaptionMode</td></tr>
<tr><td>currentSubtitleTrack</td><td>string</td><td></td><td>READ_ONLY</td><td>The identifier of the selected subtitle track. Subtitles may or may not be visible on the screen, depending upon the user's caption mode setting.

Reading this field will return the identifier of the subtitle track that is playing. When the user has not selected a track, the Roku media player will select a track based on the preferred caption language system setting.</td></tr>
<tr><td>availableSubtitleTracks</td><td>array of associative arrays</td><td>[ ] empty array</td><td>READ_ONLY</td><td>The list of subtitle tracks available in the video stream. The array is initially populated with the tracks specified in the Content Meta-Data, and additional tracks are added if they are detected by the digital video player. Each associative array has the following entries:

<table><tr><th>Key</th><th>Type</th><th>Value</th></tr><tr><td>Description</td><td>string</td><td>Descriptive name of the subtitle track</td></tr><tr><td>Language</td><td>string</td><td>ISO 639-2 three-letter language code</td></tr><tr><td>TrackName</td><td>string</td><td>The track identifier. The value of this field may be used to select the subtitle track.</td></tr></table></td></tr>
</table>

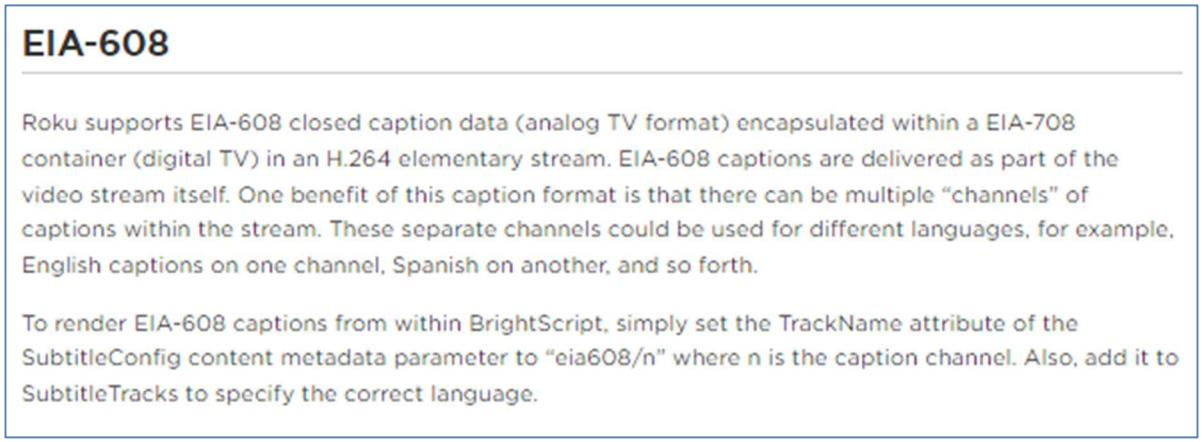
<table>
<tr>
<td>captionStyle</td>
<td>associative array</td>
<td>system default</td>
<td>READ_WRITE</td>
<td>
Available since Roku OS 8.

Allows channels to style closed captions. For any keys that are absent from the associative array, or for unexpected values, the Default value is assumed for that property. Following are the possible key names and values for this field:

<table>
<tr><th>Property</th><th>Possible Values</th></tr>
<tr><td>Text/Font</td><td>Default
Serif Fixed Width
Serif Proportional
Sans Serif Fixed Width
Sans Serif Proportional
Casual
Cursive
Small Caps</td></tr>
<tr><td>Text/Effect</td><td>Default
None
Raised
Depressed
Uniform
Drop shadow (left)
Drop shadow (right)</td></tr>
<tr><td>Text/Size</td><td>Default
Large
Medium
Small</td></tr>
<tr><td>Text/Color</td><td>Default
White
Black
Red
Green
Blue
Yellow
Magenta
Cyan</td></tr>
<tr><td>Text/Opacity</td><td>Default
25%
50%
75%
100%</td></tr>
<tr><td>Background/Color</td><td>Default
White
Black
Red
Green
Blue
Yellow
Magenta
Cyan</td></tr>
<tr><td>Background/Opacity</td><td>Default
Off
25%
50%
75%
100%</td></tr>
<tr><td>Window/Color</td><td>Default
White
Black
Red
Green
Blue
Yellow
Magenta
Cyan</td></tr>
<tr><td>Window/Opacity</td><td>Default
Off
25%
50%
75%
100%</td></tr>
</table>
</td>
</tr>
</table>

See e.g., https://developer.roku.com/en-gb/docs/references/scenegraph/media-playback-nodes/video.md#closed-caption-fields

95. On information and belief, one or more components of the Roku system employ and provide a method of preparing a received content element for presentation comprising the step of associating the content element with a container, wherein associating the content element of a container compromises reading a logical reference element of the container.

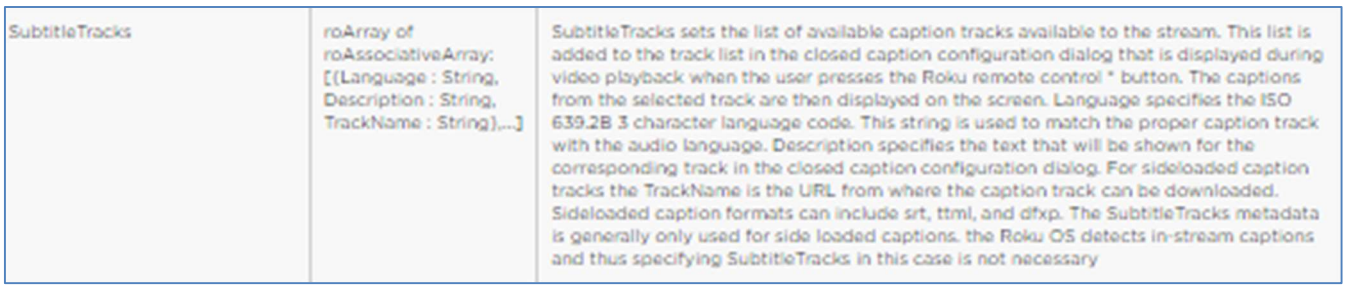


EIA-608

Roku supports EIA-608 closed caption data (analog TV format) encapsulated within a EIA-708 container (digital TV) in an H.264 elementary stream. EIA-608 captions are delivered as part of the video stream itself. One benefit of this caption format is that there can be multiple "channels" of captions within the stream. These separate channels could be used for different languages, for example, English captions on one channel, Spanish on another, and so forth.

To render EIA-608 captions from within BrightScript, simply set the TrackName attribute of the SubtitleConfig content metadata parameter to "eia608/n" where n is the caption channel. Also, add it to SubtitleTracks to specify the correct language.

See e.g., https://developer.roku.com/en-gb/docs/developer-program/media-playback/closed-caption.md.



SubtitleTracks	roArray of roAssociativeArray: [{Language : String, Description : String, TrackName : String},...]	SubtitleTracks sets the list of available caption tracks available to the stream. This list is added to the track list in the closed caption configuration dialog that is displayed during video playback when the user presses the Roku remote control * button. The captions from the selected track are then displayed on the screen. Language specifies the ISO 639.2B 3 character language code. This string is used to match the proper caption track with the audio language. Description specifies the text that will be shown for the corresponding track in the closed caption configuration dialog. For sideloaded caption tracks the TrackName is the URL from where the caption track can be downloaded. Sideloaded caption formats can include srt, ttml, and dfxp. The SubtitleTracks metadata is generally only used for side loaded captions. the Roku OS detects in-stream captions and thus specifying SubtitleTracks in this case is not necessary

See e.g., https://developer.roku.com/en-gb/docs/developer-program/getting-started/architecture/content-metadata.md.

availableSubtitleTracks	array of associative arrays	[ ] empty array	READ_ONLY	The list of subtitle tracks available in the video stream. The array is initially populated with the tracks specified in the Content Meta-Data, and additional tracks are added if they are detected by the digital video player. Each associative array has the following entries:

Key	Type	Value
Description	string	Descriptive name of the subtitle track
Language	string	ISO 639-2 three-letter language code
TrackName	string	The track identifier. The value of this field may be used to select the subtitle track.

See e.g., https://developer.roku.com/en-gb/docs/references/scenegraph/media-playback-nodes/video.md#closed-caption-fields.

96. On information and belief, one or more components of the Roku system employ and provide a method of preparing a received content element for presentation comprising the step of associating the content element with a container, wherein associating the content element of a container compromises recovering the content element in the received data corresponding to the logical reference element.

EIA-608

Roku supports EIA-608 closed caption data (analog TV format) encapsulated within a EIA-708 container (digital TV) in an H.264 elementary stream. EIA-608 captions are delivered as part of the video stream itself. One benefit of this caption format is that there can be multiple "channels" of captions within the stream. These separate channels could be used for different languages, for example, English captions on one channel, Spanish on another, and so forth.

To render EIA-608 captions from within BrightScript, simply set the TrackName attribute of the SubtitleConfig content metadata parameter to "eia608/n" where n is the caption channel. Also, add it to SubtitleTracks to specify the correct language.

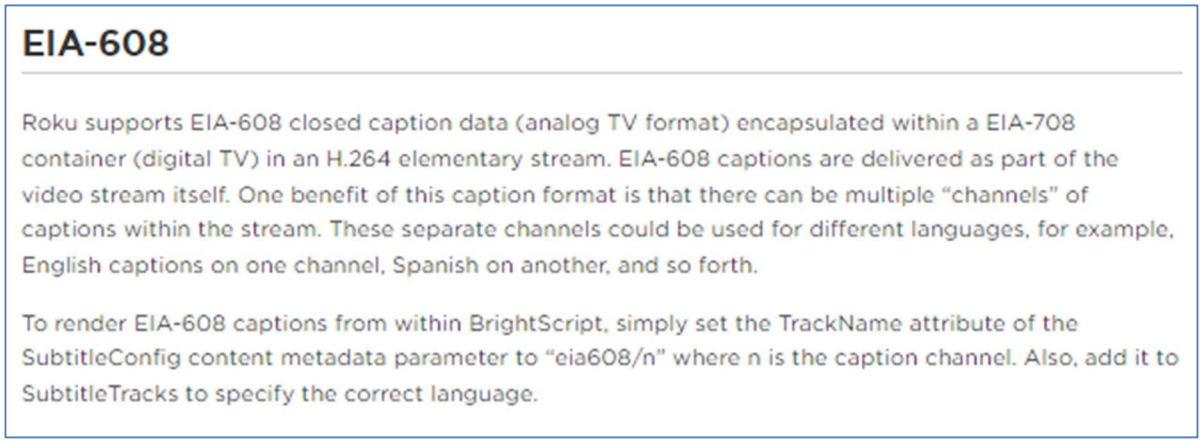
See e.g., https://developer.roku.com/en-gb/docs/developer-program/media-playback/closed-caption.md.

availableSubtitleTracks	array of associative arrays	[ ] empty array	READ_ONLY	The list of subtitle tracks available in the video stream. The array is initially populated with the tracks specified in the Content Meta-Data, and additional tracks are added if they are detected by the digital video player. Each associative array has the following entries:

Key	Type	Value
Description	string	Descriptive name of the subtitle track
Language	string	ISO 639-2 three-letter language code
TrackName	string	The track identifier. The value of this field may be used to select the subtitle track.

See e.g., https://developer.roku.com/en-gb/docs/references/scenegraph/media-playback-nodes/video.md#closed-caption-fields.

97. On information and belief, one or more components of the Roku system employ and provide a method of preparing a received content element for presentation comprising the step of associating the content element with a container, wherein associating the content element of a container compromises linking the content element with the container.

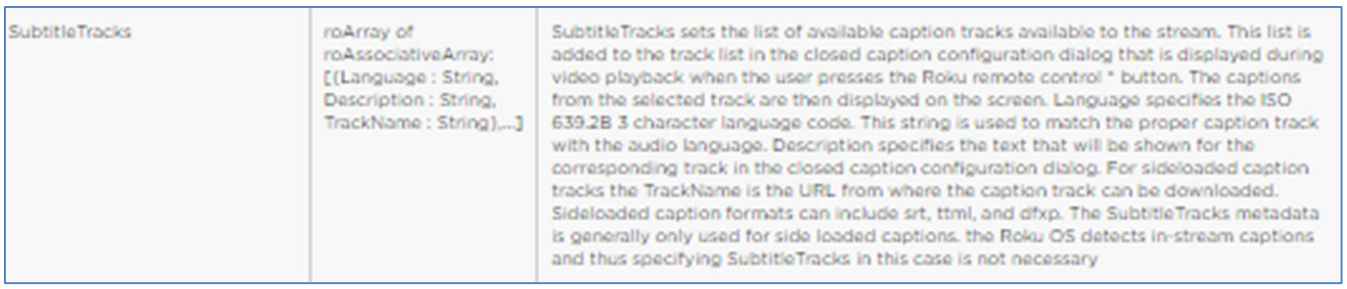


EIA-608

Roku supports EIA-608 closed caption data (analog TV format) encapsulated within a EIA-708 container (digital TV) in an H.264 elementary stream. EIA-608 captions are delivered as part of the video stream itself. One benefit of this caption format is that there can be multiple "channels" of captions within the stream. These separate channels could be used for different languages, for example, English captions on one channel, Spanish on another, and so forth.

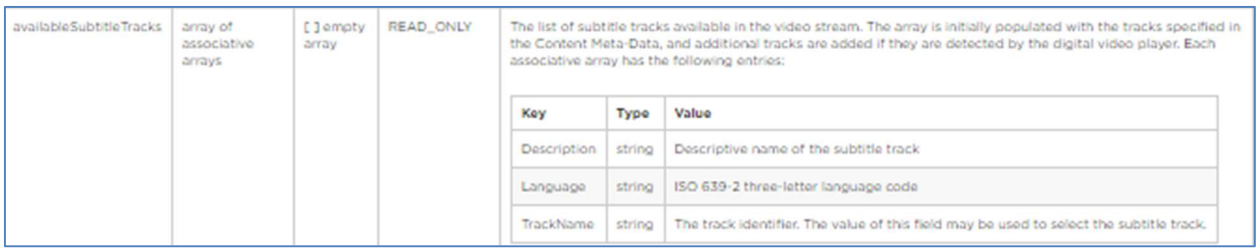
To render EIA-608 captions from within BrightScript, simply set the TrackName attribute of the SubtitleConfig content metadata parameter to "eia608/n" where n is the caption channel. Also, add it to SubtitleTracks to specify the correct language.

See e.g., https://developer.roku.com/en-gb/docs/developer-program/media-playback/closed-caption.md.



SubtitleTracks	roArray of roAssociativeArray: [{Language : String, Description : String, TrackName : String},...]	SubtitleTracks sets the list of available caption tracks available to the stream. This list is added to the track list in the closed caption configuration dialog that is displayed during video playback when the user presses the Roku remote control * button. The captions from the selected track are then displayed on the screen. Language specifies the ISO 639.2B 3 character language code. This string is used to match the proper caption track with the audio language. Description specifies the text that will be shown for the corresponding track in the closed caption configuration dialog. For sideloaded caption tracks the TrackName is the URL from where the caption track can be downloaded. Sideloaded caption formats can include srt, ttml, and dfxp. The SubtitleTracks metadata is generally only used for side loaded captions. the Roku OS detects in-stream captions and thus specifying SubtitleTracks in this case is not necessary

See e.g., https://developer.roku.com/en-gb/docs/developer-program/getting-started/architecture/content-metadata.md.



availableSubtitleTracks	array of associative arrays	[ ] empty array	READ_ONLY	The list of subtitle tracks available in the video stream. The array is initially populated with the tracks specified in the Content Meta-Data, and additional tracks are added if they are detected by the digital video player. Each associative array has the following entries:

Key	Type	Value
Description	string	Descriptive name of the subtitle track
Language	string	ISO 639-2 three-letter language code
TrackName	string	The track identifier. The value of this field may be used to select the subtitle track.

See e.g., https://developer.roku.com/en-gb/docs/references/scenegraph/media-playback-nodes/video.md#closed-caption-fields.

98. Roku's direct infringement has damaged AdaptFlow and caused it to suffer and continue to suffer irreparable harm and damages.

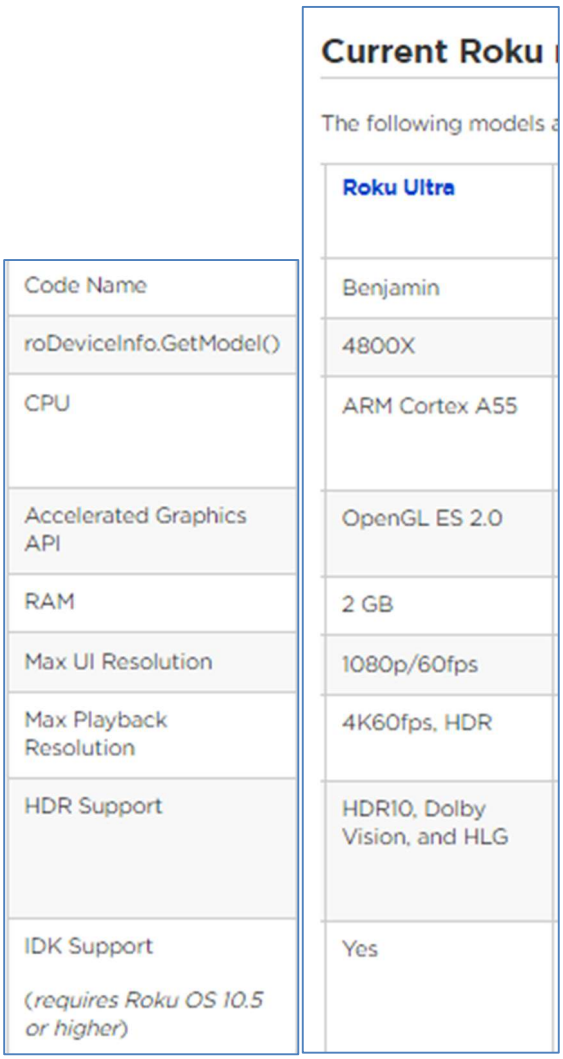
Count IV – Infringement of United States Patent No. 7,539,012

99. AdaptFlow repeats, realleges, and incorporates by reference, as if fully set forth here, the allegations of the preceding paragraphs above.

100. On information and belief, Roku (or those acting on its behalf) makes, uses, sells, imports and/or offers to sell the Roku Products and Services; and makes, uses, sells, sells access to, imports, offers to sell and/or offers to sell access to the Roku System in the United States that infringe (literally and/or under the doctrine of equivalents) at least claim 1 of the '012 patent.

101. On information and belief, one or more components of the Roku system include a data storage medium read/write unit, that is designed to be installed in an electronic apparatus.

See e.g., https://www.roku.com/products/roku-ultra.



Current Roku

The following models

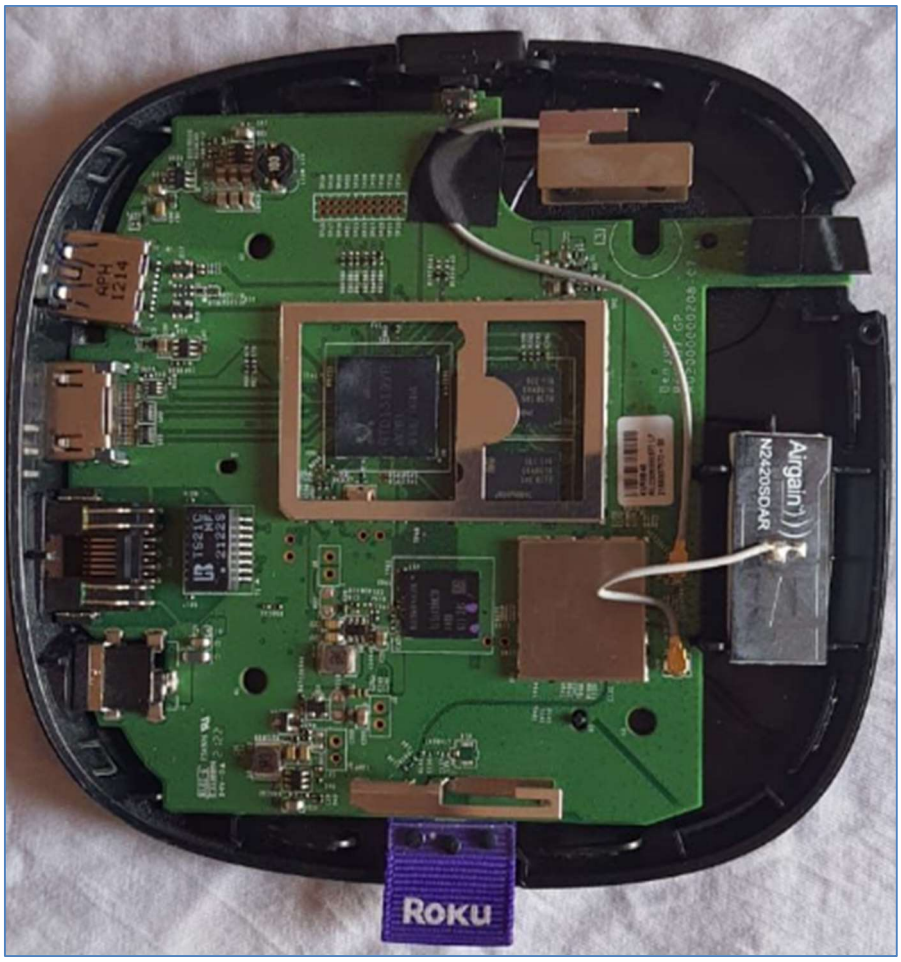
	Roku Ultra
Code Name	Benjamin
roDeviceInfo.GetModel()	4800X
CPU	ARM Cortex A55
Accelerated Graphics API	OpenGL ES 2.0
RAM	2 GB
Max UI Resolution	1080p/60fps
Max Playback Resolution	4K60fps, HDR
HDR Support	HDR10, Dolby Vision, and HLG
IDK Support *(requires Roku OS 10.5 or higher)*	Yes

See e.g., https://developer.roku.com/en-ca/docs/specs/hardware.md.

The first difference evident between this and previous generations of Roku devices is the heat sinking. Older Roku boxes had an aluminum piece screwed into the top half of the case. It provided a large heat-sink that cooled the main processor. The aluminum piece is gone from the new Ultra. Instead, designers used a smaller two-piece heat sink that fits on top of the metal shielding that protects the two SDRAM chips and the main display processor. Thermal grease connects the display processor to the heat sinking.

> The Realtek display processor sits in a metal-shielded area together with two 8 Gb SDRAM chips from Nanya, a Taiwanese memory maker.

See e.g., https://www.microcontrollertips.com/teardown-the-new-roku-ultra/.



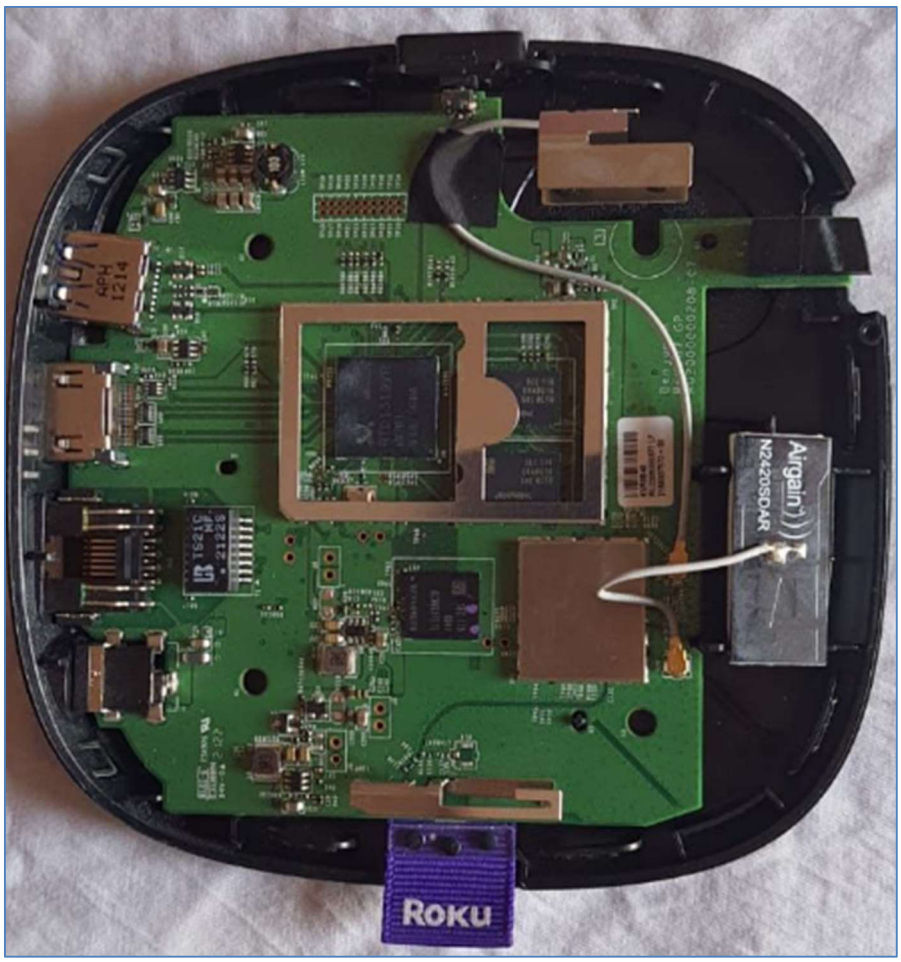
See e.g., images of teardown of Roku Ultra.

102. On information and belief, one or more components of the Roku system include an electronic read/write device.

The first difference evident between this and previous generations of Roku devices is the heat sinking. Older Roku boxes had an aluminum piece screwed into the top half of the case. It provided a large heat-sink that cooled the main processor. The aluminum piece is gone from the new Ultra. Instead, designers used a smaller two-piece heat sink that fits on top of the metal shielding that protects the two SDRAM chips and the main display processor. Thermal grease connects the display processor to the heat sinking.

The Realtek display processor sits in a metal-shielded area together with two 8 Gb SDRAM chips from Nanya, a Taiwanese memory maker.

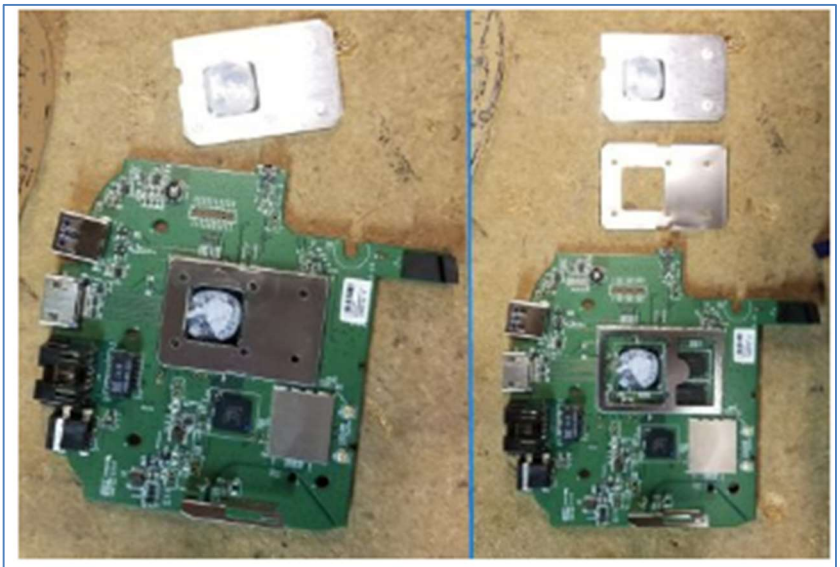
See e.g., https://www.microcontrollertips.com/teardown-the-new-roku-ultra/.



See e.g., images of teardown of the Roku Ultra.

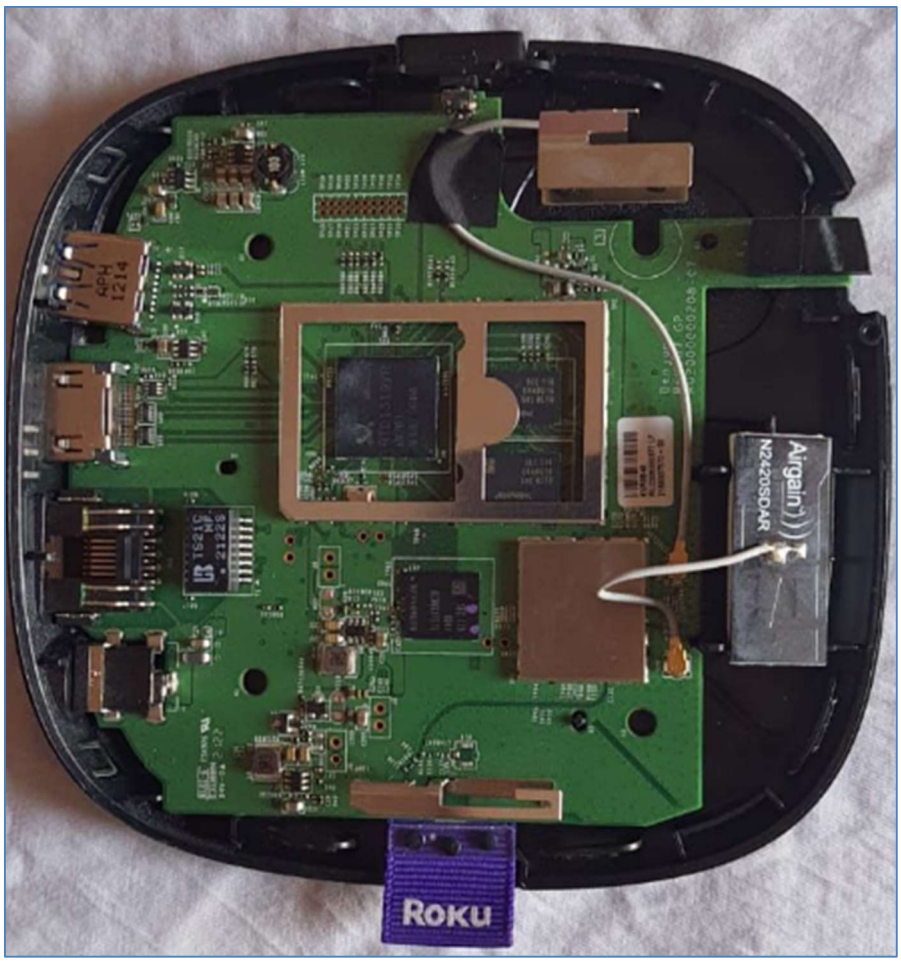
103. On information and belief, one or more components of the Roku system include a plastic, thermal non-conducting housing definitively bound to the electronic read/write device, so that the opening of the housing, and housing sides are assembled in a permanent fashion.


Roku
Roku
OK
1
2
hulu
VUDU
Roku



The two-piece heat sink in stages of disassembly. Click image to enlarge.

See e.g., https://www.microcontrollertips.com/teardown-the-new-roku-ultra/.



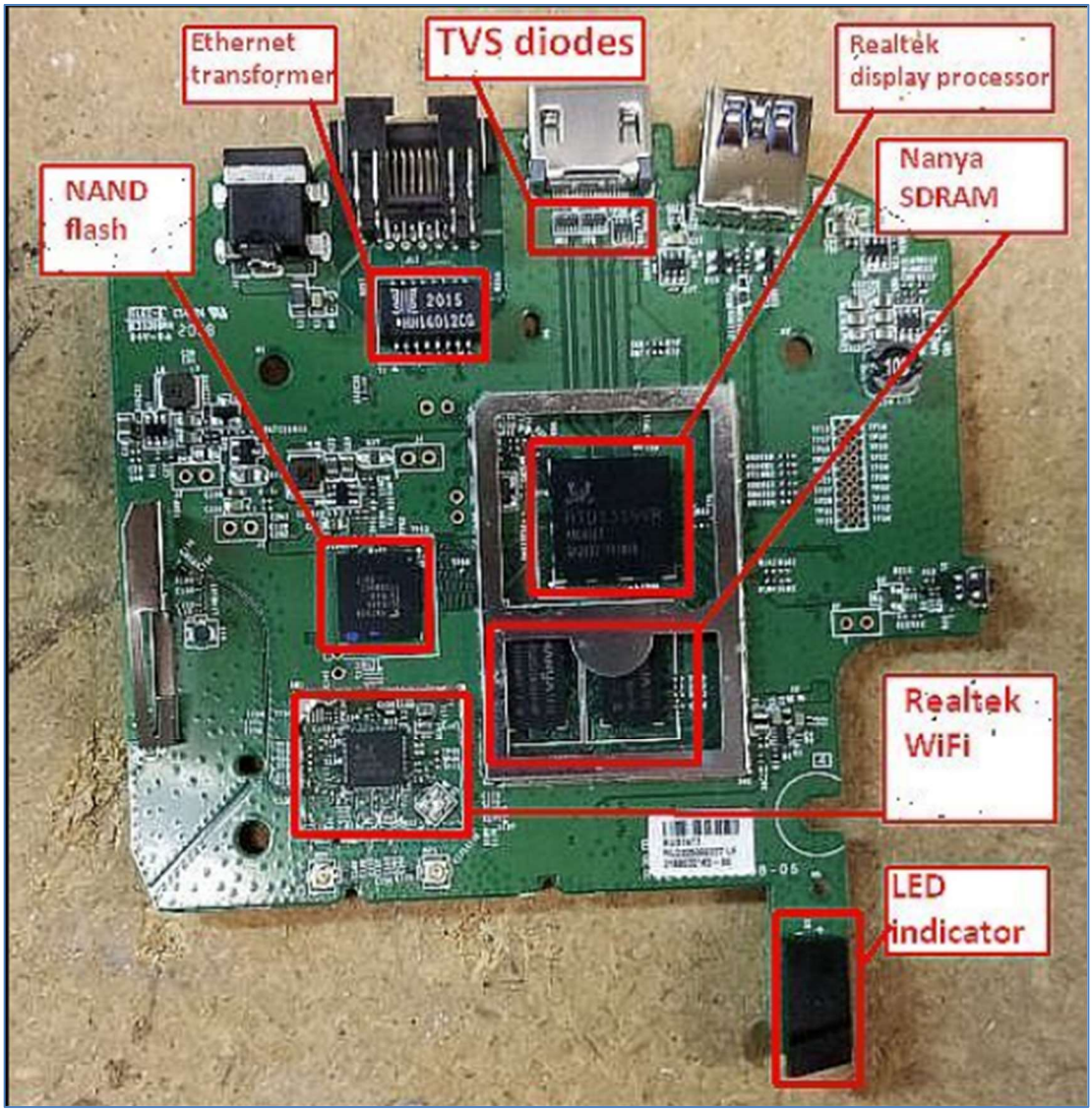
See e.g., images of teardown of the Roku Ultra.

- Do not drop, crush or disassemble the Player.
- Do not attempt to repair your Player yourself. Disassembling the Player may cause damage not covered by the warranty.

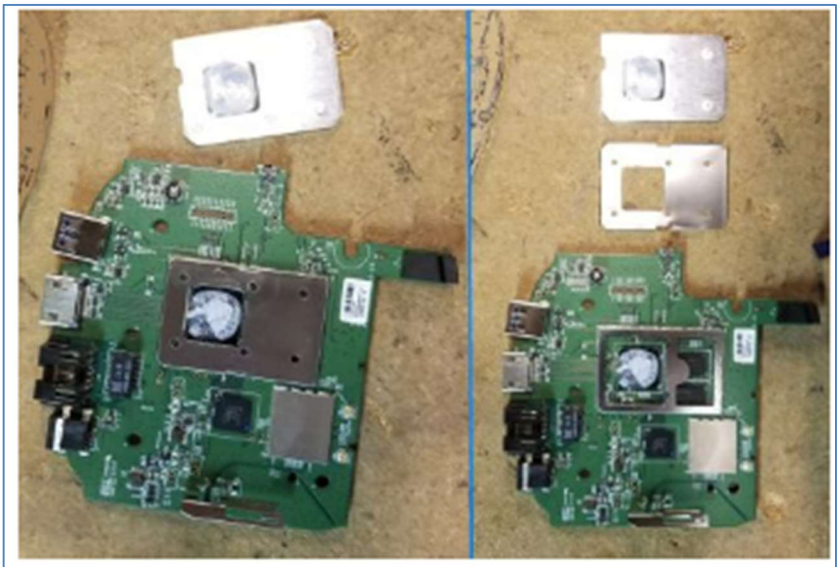
See e.g., https://manuals.plus/roku/roku-streaming-player-information-manual#axzz7avG84OzJ.

104. On information and belief, one or more components of the Roku system employ and provide a method that contains a housing where at least one embedded

metallic heat sink is closely in contact with the lateral sides of the electronic read/write device for cooling it.

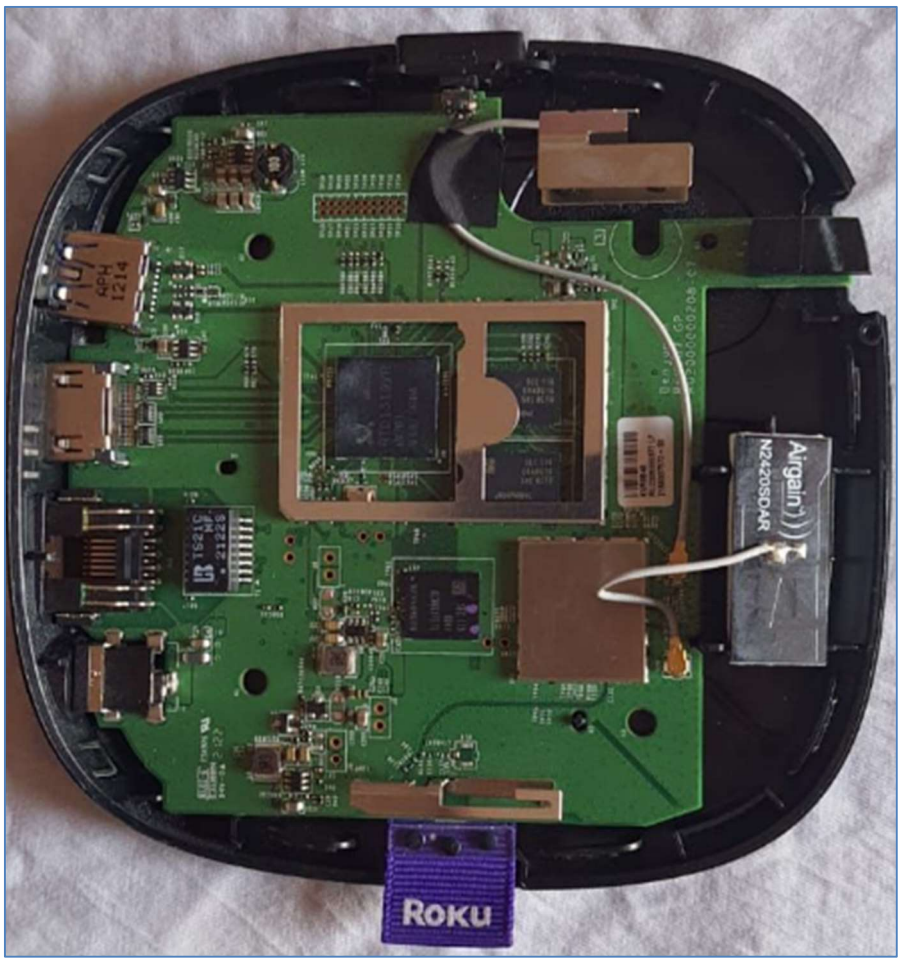


See e.g., https://www.microcontrollertips.com/wp-content/uploads/2020/12/PCB-top-and-bottom.jpg.



The two-piece heat sink in stages of disassembly. Click image to enlarge.

See e.g., https://www.microcontrollertips.com/teardown-the-new-roku-ultra.



See e.g., images of teardown of the Roku Ultra.

105. Roku's direct infringement has damaged AdaptFlow and caused it to suffer and continue to suffer irreparable harm and damages.

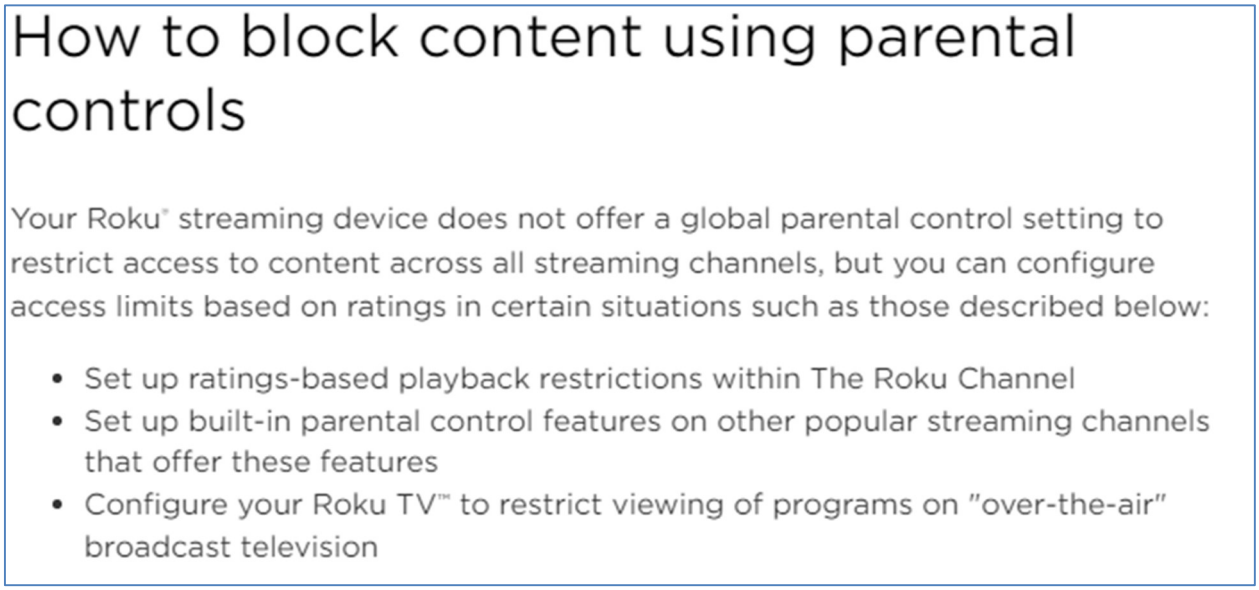
Count V – Infringement of United States Patent No. 8,447,162

106. AdaptFlow repeats, realleges, and incorporates by reference, as if fully set forth here, the allegations of the preceding paragraphs above.

107. On information and belief, Roku (or those acting on its behalf) makes, uses, sells, imports and/or offers to sell the Roku Products and Services; and makes, uses, sells, sells access to, imports, offers to sell and/or offers to sell access to the Roku

System in the United States that infringe (literally and/or under the doctrine of equivalents) at least claim 6 of the '162 patent.

108. On information and belief, one or more components of the Roku system employ and provide a method comprising the receiving of a command and saving a first set of settings of a controller for a receiver in response to the command.



How to block content using parental controls

Your Roku® streaming device does not offer a global parental control setting to restrict access to content across all streaming channels, but you can configure access limits based on ratings in certain situations such as those described below:

- Set up ratings-based playback restrictions within The Roku Channel
- Set up built-in parental control features on other popular streaming channels that offer these features
- Configure your Roku TV™ to restrict viewing of programs on "over-the-air" broadcast television

See e.g., https://support.roku.com/article/208755938.

Parental controls on The Roku Channel

The Roku Channel parental controls is a PIN-based playback control that applies only when you are streaming content on The Roku Channel in a signed-in state. When this feature is enabled, your PIN must be entered when someone tries watching content with certain ratings on The Roku Channel. What content can be played without a PIN depends on which of the four ratings levels you select when configuring your PIN preferences, as described in the table below.

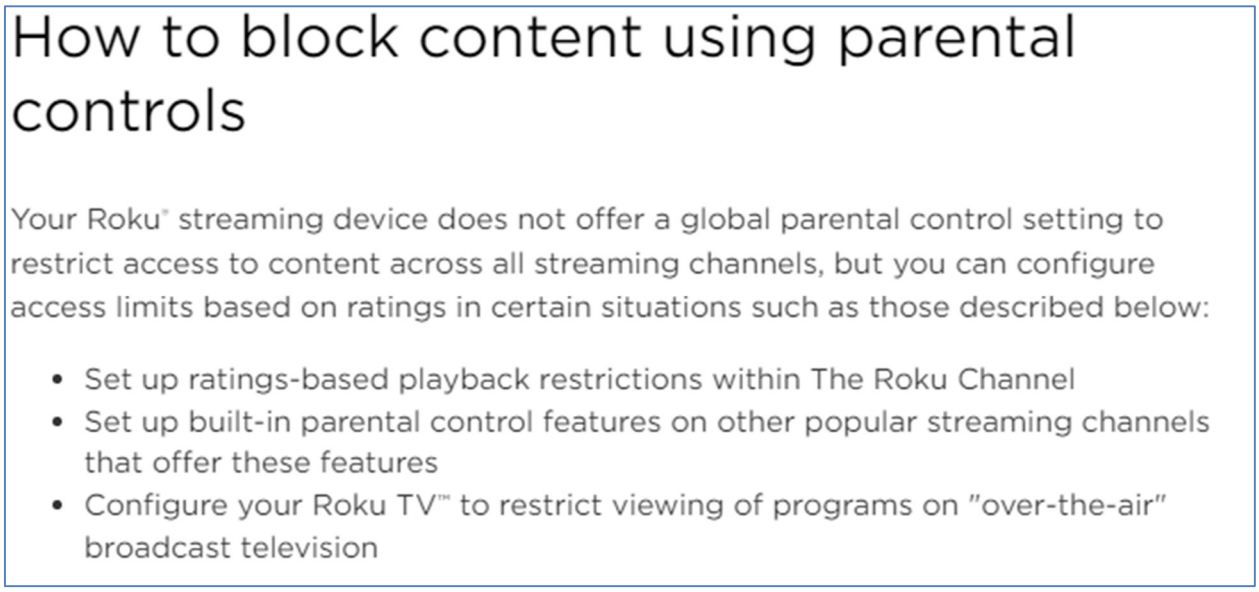
✓ = Content with this rating on The Roku Channel can be watched without a PIN

✗ = A PIN is required to watch content with this rating on The Roku Channel

Ratings	Off	G, TV-Y, TV-G	PG, TV-PG	PG-13, TV-14
G	✓	✓	✓	✓
TV-Y	✓	✓	✓	✓
TV-G	✓	✓	✓	✓
TV-Y7	✓	✗	✓	✓
TV-PG	✓	✗	✓	✓
PG	✓	✗	✓	✓
TV-14	✓	✗	✗	✓
PG-13	✓	✗	✗	✓
TV-MA	✓	✗	✗	✗
R	✓	✗	✗	✗
NC-17	✓	✗	✗	✗
Unrated	✓	✗	✗	✗
Not Rated	✓	✗	✗	✗

See e.g., https://www.roku.com/features/parental-controls/.

109. On information and belief one or more components of the Roku system employ and provide a method comprising loading a second set of settings associated with the command into the controller after saving the first set of settings.

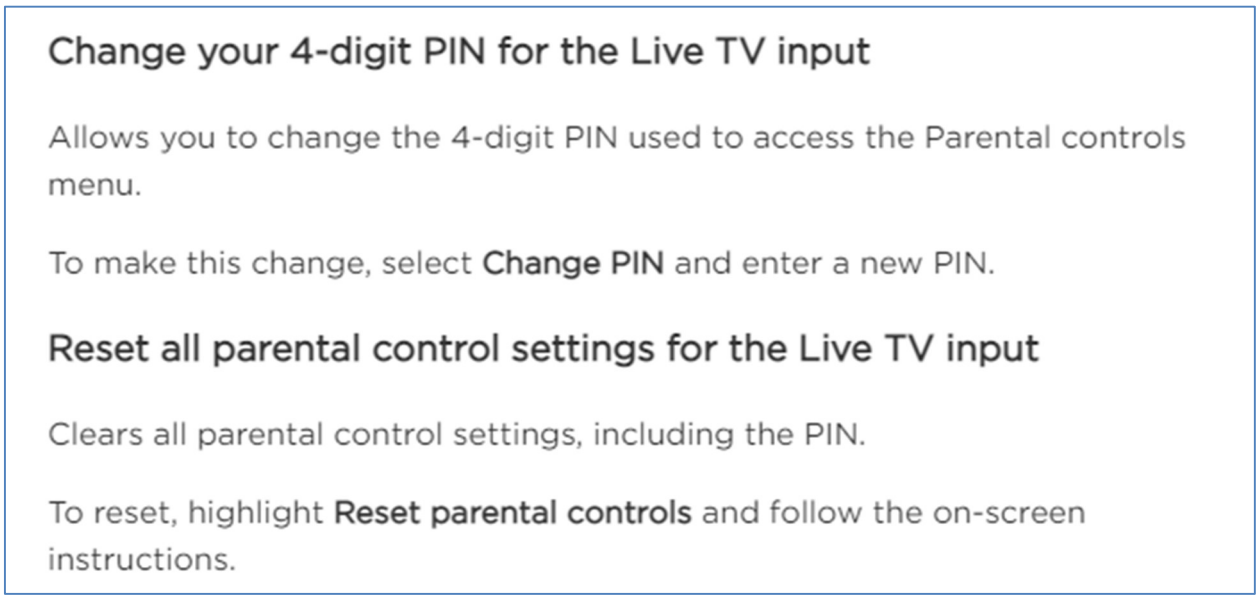


How to block content using parental controls

Your Roku® streaming device does not offer a global parental control setting to restrict access to content across all streaming channels, but you can configure access limits based on ratings in certain situations such as those described below:

- Set up ratings-based playback restrictions within The Roku Channel
- Set up built-in parental control features on other popular streaming channels that offer these features
- Configure your Roku TV™ to restrict viewing of programs on "over-the-air" broadcast television

See e.g., https://support.roku.com/article/208755938.



Change your 4-digit PIN for the Live TV input

Allows you to change the 4-digit PIN used to access the Parental controls menu.

To make this change, select **Change PIN** and enter a new PIN.

Reset all parental control settings for the Live TV input

Clears all parental control settings, including the PIN.

To reset, highlight **Reset parental controls** and follow the on-screen instructions.

See e.g., https://support.roku.com/article/208755938.

How to set a Parental Controls PIN for The Roku Channel

A Parental Controls PIN uses MPAA and TV ratings to restrict playback of select content within The Roku Channel. It does not hide, filter, or affect playback outside The Roku Channel; nor does it prevent users from accessing content on other streaming channels or platforms (such a provider's website or mobile app).

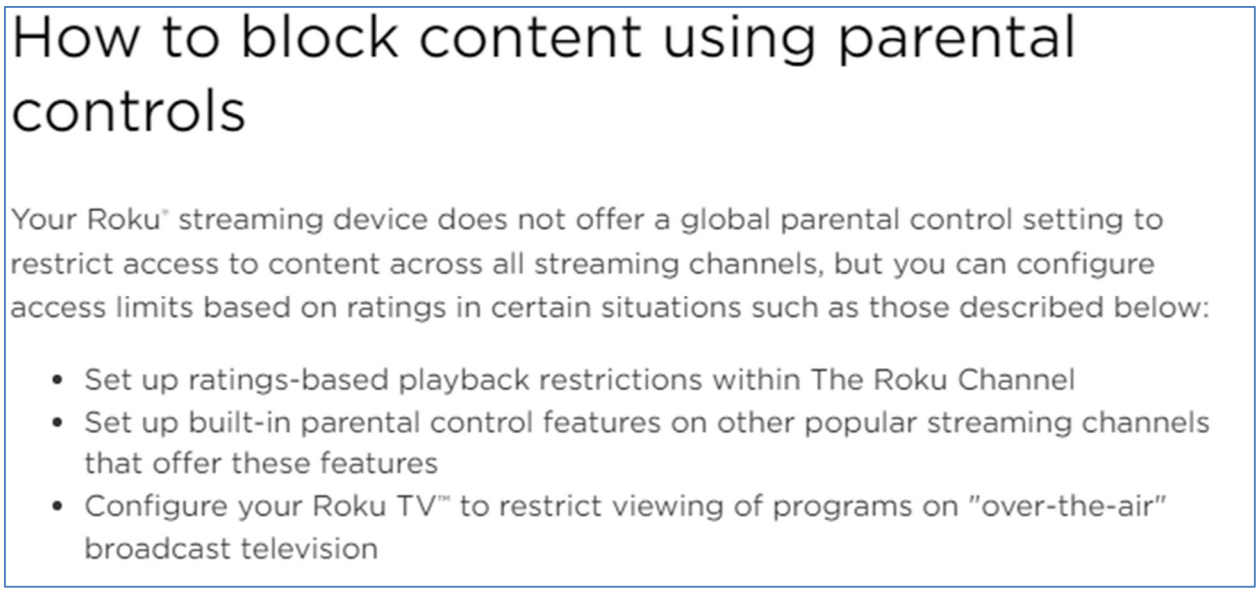
If you choose to use a PIN for parental controls, you will enter it the first time you access restricted content after turning on your Roku device.

Once you access your PIN preferences, follow the steps below to set a parental controls PIN.

1. Go to *Parental Controls for The Roku Channel* and select any of the following:
   - **Off**
   - **G, TV-Y, TV-G** - You can watch content rated G, TV-Y, and TV-G without requiring a PIN. Best for little kids.
   - **PG, TV-PG** -You can watch content rated PG, TV-PG, and below without requiring a PIN. Best for young kids.
   - **PG-13, TV-14** - You can watch content rated PG-13, TV-14, and below without requiring a PIN. Best for teens.
2. Select **Save preferences**
3. If you are prompted to create a PIN, enter the four-digit PIN you wish to use and select **Save PIN**

See e.g., https://support.roku.com/article/208755938.

110. On information and belief, one or more components of the Roku system employ and provide a method comprising receiving setting changes to the second set of settings from a user.



See e.g., https://support.roku.com/article/208755938.

Parental controls on The Roku Channel

The Roku Channel parental controls is a PIN-based playback control that applies only when you are streaming content on The Roku Channel in a signed-in state. When this feature is enabled, your PIN must be entered when someone tries watching content with certain ratings on The Roku Channel. What content can be played without a PIN depends on which of the four ratings levels you select when configuring your PIN preferences, as described in the table below.

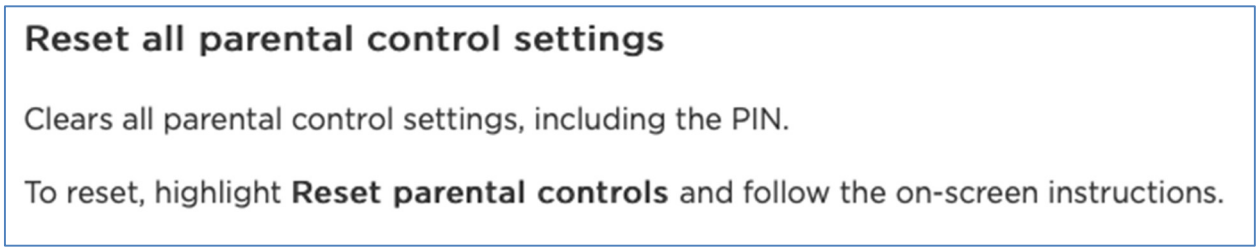
✓ = Content with this rating on The Roku Channel can be watched without a PIN

✗ = A PIN is required to watch content with this rating on The Roku Channel

Ratings	Off	G, TV-Y, TV-G	PG, TV-PG	PG-13, TV-14
G	✓	✓	✓	✓
TV-Y	✓	✓	✓	✓
TV-G	✓	✓	✓	✓
TV-Y7	✓	✗	✓	✓
TV-PG	✓	✗	✓	✓
PG	✓	✗	✓	✓
TV-14	✓	✗	✗	✓
PG-13	✓	✗	✗	✓
TV-MA	✓	✗	✗	✗
R	✓	✗	✗	✗
NC-17	✓	✗	✗	✗
Unrated	✓	✗	✗	✗
Not Rated	✓	✗	✗	✗

See e.g., https://www.roku.com/features/parental-controls/.

111. On information and belief one or more components of the Roku system employs and provides a method comprising reloading unchanged settings from the first set of setting into the controller when the command is no longer valid.



See e.g., https://support.roku.com/article/208755938.

112. Roku's direct infringement has damaged AdaptFlow and caused it to suffer and continue to suffer irreparable harm and damages.

Count VI – Infringement of United States Patent No. 8,667,068

113. AdaptFlow repeats, realleges, and incorporates by reference, as if fully set forth here, the allegations of the preceding paragraphs above.

114. On information and belief, Roku (or those acting on its behalf) makes, uses, sells, imports and/or offers to sell the Roku Products and Services; and makes, uses, sells, sells access to, imports, offers to sell and/or offers to sell access to the Roku System in the United States that infringe (literally and/or under the doctrine of equivalents) at least claim 1 of the '068 patent.

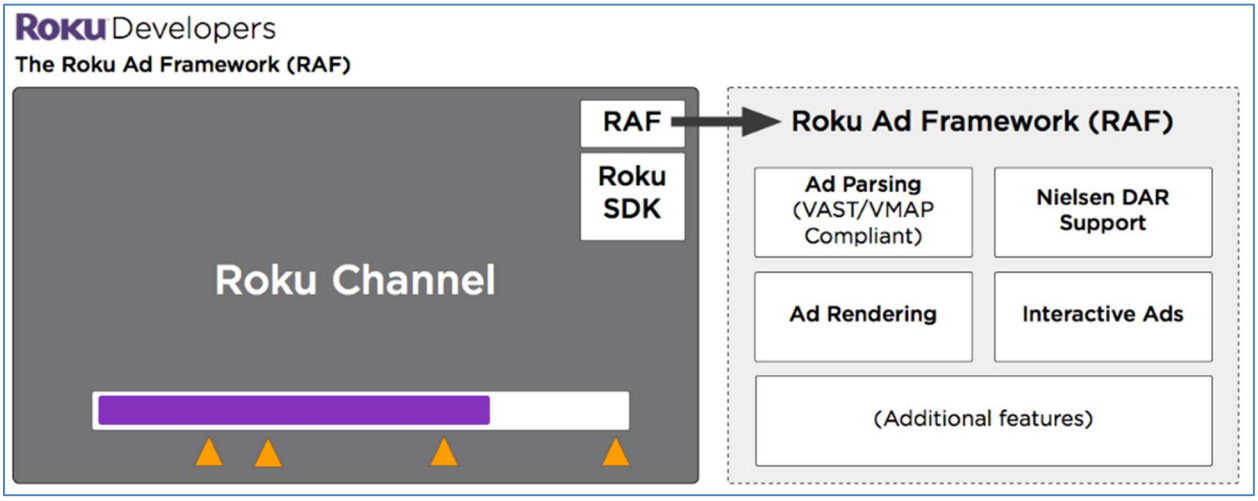
115. On information and belief, one or more components of the Roku system employ and provide a method for delivering an electronic message to an audience at an entertainment venue.

Roku Advertising Framework overview

The Roku Advertising Framework (RAF) enables the seamless integration of video advertising into your channels. The RAF library, which is built directly into the Roku SDK, includes the following features that make it easy to provide a consistent ad experience across channels:

- Parsing of ads in VAST2, VAST3, VMAP, and FreeWheel's SmartXML formats (see the table below for details).
- Built-in solution for displaying client-side (CSAI) video ads that works with Google Ad Manager (formerly known as DFP), FreeWheel, SpotX, and other 3rd-party servers.
- Playback control for server-stitched ads.
- Client-side handling of tracking events that is aligned with the IAB/MRC's impression measurement guidelines.
- Audience measurement via Nielsen DAR/DCR, Comscore vCE, and other platforms.
- Interactive ads through Innovid, BrightLine, and Roku.
- Client-side solutions to minimize buffering between ads and content.
- Samples for implementing server-side ad insertion (SSAI) with Verizon Media Services, Adobe, Brightcove, Yospace, AWS Elemental MediaTailor servers, and Google Ad Manager Dynamic Ad Insertion (DAI).

See e.g., https://developer.roku.com/en-ca/docs/developer-program/advertising/roku-advertising-framework.md.



Features

- Automatic appending of key targeting parameters in the ad request (Roku ID for Advertising, content genre, display dimensions, etc.)
- IAB VAST 2.0/3.0 parsing
- IAB VMAP parsing
- FreeWheel SmartXML[5] parsing
- Interactive ad rendering
- Audience measurement

For the full RAF integration cases, the samples:

1. Initializes RAF
2. Turns on Nielsen tracking, and configures it with genre, program ID, and content parameters
3. Configures the ad URL using URL macros (See URL Parameter Macros in Integrating the Roku Advertising Framework) and the `setAdUrl()` method
4. Gets the ads using getAds() (VAST feed), and renders them using showAds()

For non-standard ad responses, ads are imported from a non-standard feed, neither VMAP, VAST or SmartXML. RAF is configured as if for a standard feed, with backfill ads disabled and extra debug output enabled. Ads are parsed from local JSON file, then formatted as an ad pods array, and imported into RAF using the `importAds()` method. After that, the sample checks for particular ads to play by passing video playback events to the RAF `getAds()` method inside event loop. If any ads were returned from `getAds()`, they are rendered using the RAF `showAds()` method. For the Scene Graph example, after `importAds()`, the sample checks for particular ads to play by passing fake video events created with `createPlayPosMsg()` to the RAF `getAds()` method before event-loop (preroll ads) and inside it (midroll/postroll ads). If any ads were returned from `getAds()`, they are rendered using the RAF `showAds()` method.

See e.g., https://sdkdocs-archive.roku.com/Roku-Advertising-Framework_3737569.html.

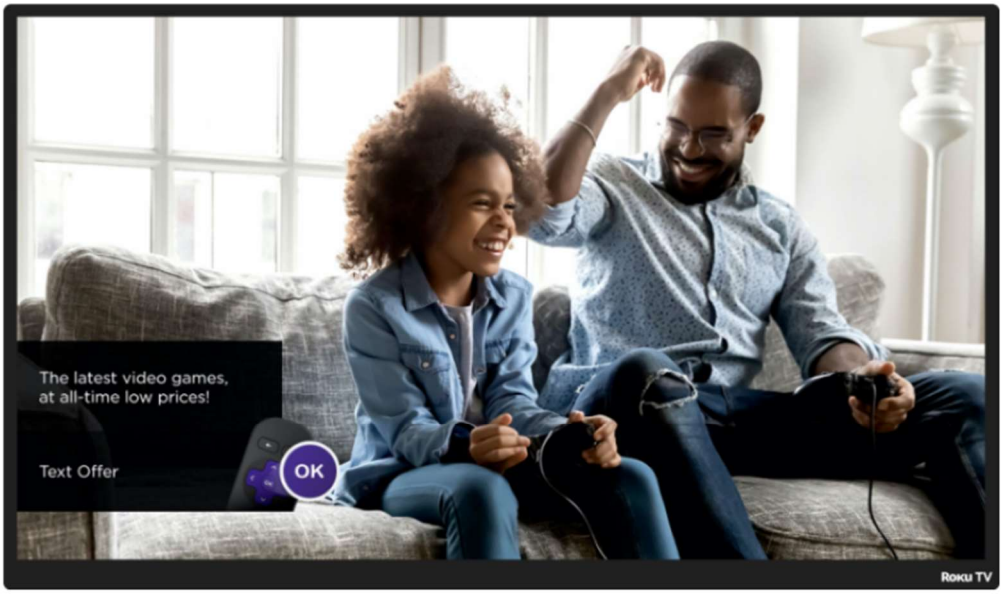
The 3 Types of Roku Interactive Ads

There are three primary ways that you can serve great, interactive ad experiences to viewers. You can also choose whether you want to run ads on both streaming and traditional TV environments, or just run across one environment.

Here are the three types of Roku Interactive Ads, with anonymized examples of what they look like in action:

1. Direct to Product.

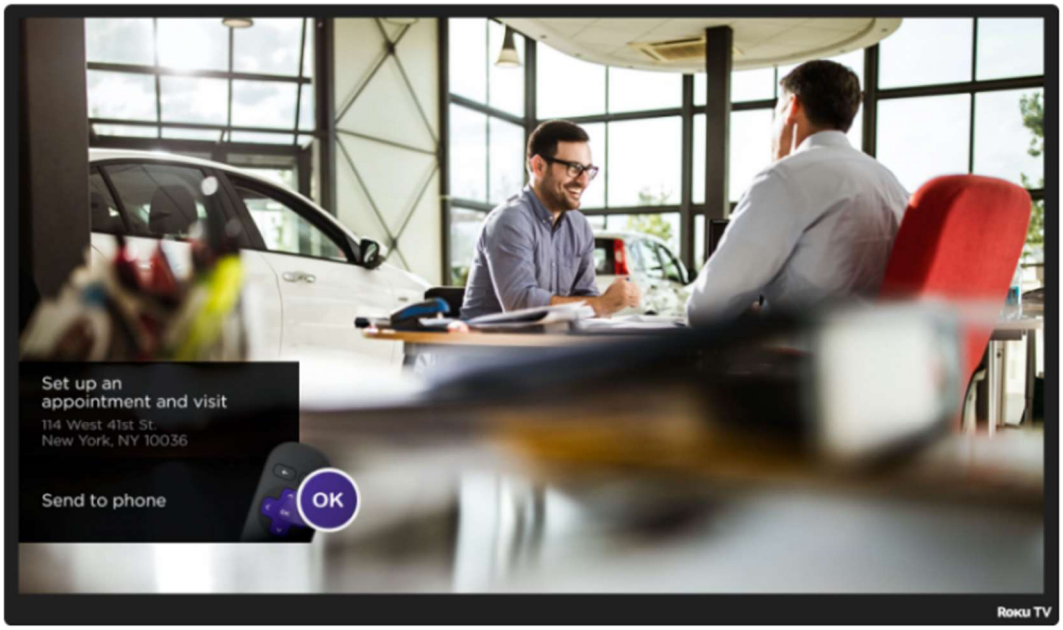
A direct-to-product interactive ad shows a product-oriented overlay on top of the creative. Advertisers on Roku frequently use Direct to Product ads to reveal a product announcement, coupon, or other offer.



In this example, a viewer would watch the video and see the offer to get more information or a coupon about specific products. The interactive ad invites viewers to share their phone numbers or enter their emails with their remotes and receive a text message and/or email that takes them to the next step.

2. Direct-to-Store

A direct-to-store interactive ad shows an overlay with the physical address of a retail location near the streamer . By using the Roku remote, streamers can add the address to their phones.



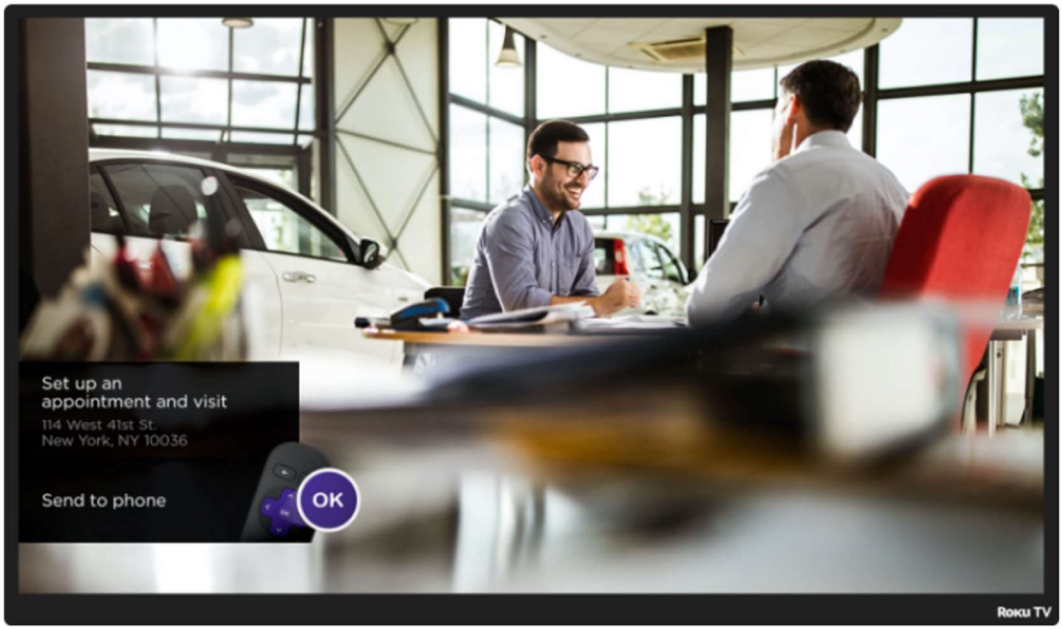
A direct-to-store interactive ad can help drive in-store traffic, show viewers important event information, or prompt them to set up appointments and visits.

See e.g., https://advertising.roku.com/resources/case-studies/roku-interactive-ads-helped-baskin-robbins-show-a.

116. On information and belief, one or more components of the Roku system employ and provide a method for delivering an electronic message to an audience at an entertainment venue comprising receiving a package containing first content for all users and associated second content for a subset of users, the package also containing an instruction file for instructing how the first content and second content are to be played.

2. Direct-to-Store

A direct-to-store interactive ad shows an overlay with the physical address of a retail location near the streamer . By using the Roku remote, streamers can add the address to their phones.

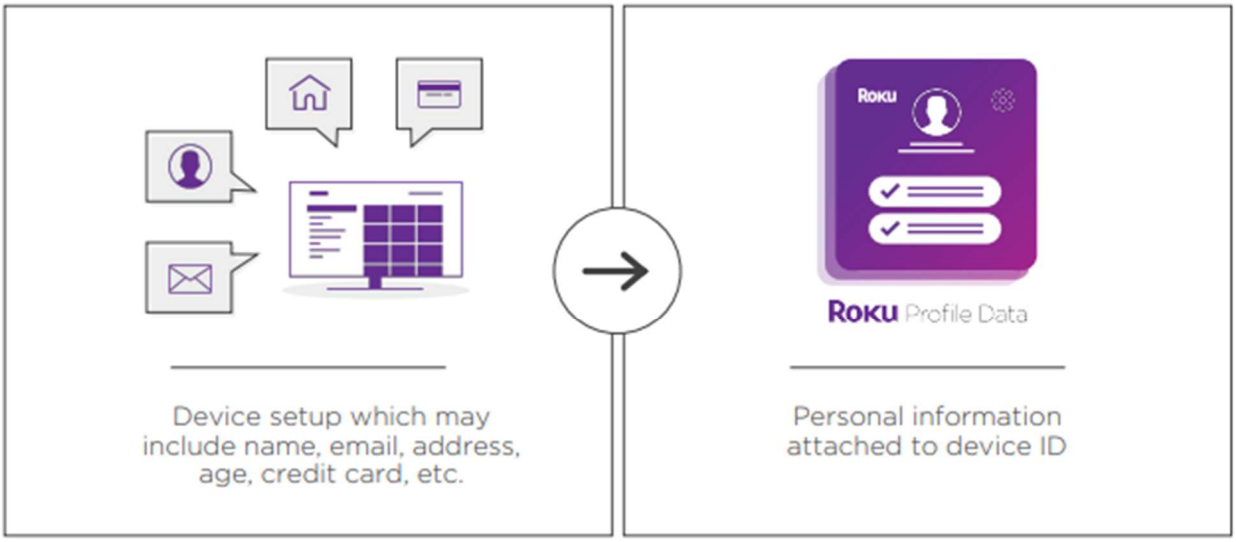


A direct-to-store interactive ad can help drive in-store traffic, show viewers important event information, or prompt them to set up appointments and visits.

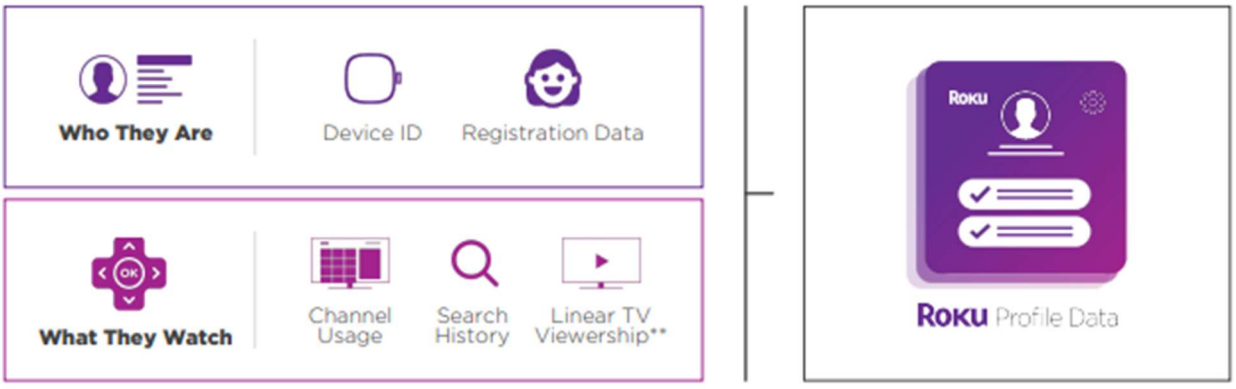
See e.g., https://advertising.roku.com/resources/case-studies/roku-interactive-ads-helped-baskin-robbins-show-a.

How Data Works

What makes Roku data so powerful? It starts with the streamer. When users set up their devices, they provide their name, email, address, date of birth, and credit card information. This is all tied to a streamer's profile on Roku.



In addition to registration data and device IDs, Roku captures viewership signals to understand which streaming channels users are watching on Roku , what they're searching and what shows they're watching on linear TV. This includes viewership captured via Automatic Content Recognition* in which the user opts in to share data.



Viewership signals allow you to build campaigns for audiences that are already watching similar content to your service and may have similar habits to your prospective viewers. We categorize viewership signals by five key categories.

General Behaviors

Audiences based on how long they have been Roku users, if they have linear TV, what types of channels and subscription services they watch, etc.

Key Value Prop: Great for reaching a broad group of users.

Linear TV Viewership*

Audiences based on engagement with linear TV content.

Key Value Prop: Great for partners looking to reach linear viewers that watched a previous season, competitor shows or networks, similar genres, etc.

Channel Usage

Audiences based on engagement within your specific streaming service.

Key Value Prop: Great for reaching streamers that have taken specific actions within your streaming service, like installing but not subscribing or viewing

Content Affinity

Audiences based on content affinity using streaming and linear viewership behaviors (i.e. comedy viewers, sports fans).

Key Value Prop: Great for reaching linear viewers and streamers that watch specific genres or content type.

Geography

Audiences based on geographic region.

Key Value Prop: Great for reaching users to watch local-specific content or grow an audience in a particular region.

Exclusion

Suppress any of the targeting audiences above (i.e. New York DMA or SVOD viewers).

Key Value Prop: Suppress specific users to attract only those that matter.

Tip: Avoid hyper-targeting because it can lead to campaign delivery issues.

See e.g., https://advertising.roku.com/publisher-solutions.

At this time last year, we reported that a Roku update included interactive pop-up ads when some users were watching cable TV or local channels with an antenna. Those interactive ads appear to be back for some Roku TV users.

The ads spotted last year included an ad for a movie on The Roku Channel, sponsored by Geico, during a Geico commercial. At the time, a Roku spokesperson commented that the ads were not new and had been active for two years.

A CCN reader sent in the following image of another type of offer this week. He says this screen has been popping up while using a PlayStation 4, offering a deal that can be redeemed via email and displaying the email connected to his Roku account.

See e.g., https://www.cordcuttersnews.com/roku-interactive-pop-up-ads-are-popping-up-again-on-roku-tvs/.

Sample Channels

The table below provides a number of samples channels to help you get started with your own RAF implementations:

Download File	SDK Version	Description
RAF Video Node Sample	SceneGraph	This sample demonstrates the use of RAF in a Video node with support for pre, mid, and post roll ads.
FullRAFSceneGraphSample.zip	SceneGraph	This sample shows how to add RAF to a channel, configure Nielsen, obtain ads and play them as pre-, mid- and post-rolls. It also shows how to import Ads from non-standard feed (neither VMAP, VAST or SMartXML.)
CustomBufferScreenSceneGraphSample.zip	SceneGraph	This sample shows how to change default ad buffering screen background, title and description. It also shows how to create a completely custom buffering screen.
FullRAFSDK1Sample.zip	Legacy SDK	This sample shows how to add RAF to a channel, configure Nielsen, obtain ads and play them as pre-, mid- and post-rolls. It also shows how to import Ads from non-standard feed (neither VMAP, VAST or SMartXML.) This sample also contains examples of custom buffering screens and stitched ads.
CustomBufferingScreenSDK1Sample.zip	Legacy SDK	This sample shows how to change default ad buffering screen background, title and description. It also shows how to create a completely custom buffering screen.
ServerStitchedAdSDK1Sample.zip	Legacy SDK	This sample demonstrates the interactive stitched ads feature. It shows how to configure RAF with interactive ads and process interactive ad events.

See e.g., https://sdkdocs-archive.roku.com/Roku-Advertising-Framework_3737569.html.

Tracking

Tracking events are triggered automatically during ad rendering by showAds(). For client applications that perform their own ad rendering, the valid event types that must be handled are represented in the `tracking` array of the Ad Structure by:

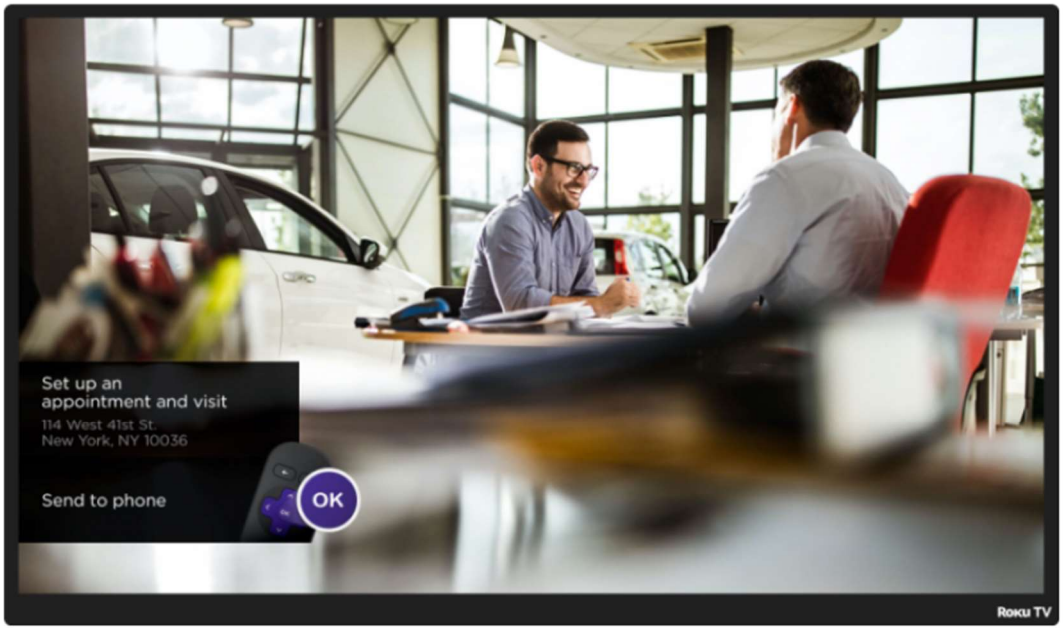
Event name	Trigger condition
Impression	Start of ad render (e.g., first frame of a video ad displayed)
PodStart	Beginning of ad pod render
PodComplete	Completed rendering ad pod
FirstQuartile	25% of video ad rendered
Midpoint	50% of video ad rendered
ThirdQuartile	75% of video ad rendered
Complete	100% of video ad rendered
Error	Error during ad parsing or rendering (VAST 3.0)
Close	User exited out of ad rendering before completion
Skip	User skipped ad (if skippable)
Pause	User paused ad
Resume	User resumed ad
Rewind	User rewound ad
Mute	User muted ad
Unmute	User un-muted ad
AcceptInvitation	User launched another portion of an ad (for interactive ads)

See e.g., https://developer.roku.com/docs/developer-program/advertising/integrating-roku-advertising-framework.md.

117. On information and belief, one or more components of the Roku system employ and provide a method for delivering an electronic message to an audience at an entertainment venue comprising separating the first content from the second content based on the instruction file.

2. Direct-to-Store

A direct-to-store interactive ad shows an overlay with the physical address of a retail location near the streamer . By using the Roku remote, streamers can add the address to their phones.



A direct-to-store interactive ad can help drive in-store traffic, show viewers important event information, or prompt them to set up appointments and visits.

See e.g., https://advertising.roku.com/resources/case-studies/roku-interactive-ads-helped-baskin-robbins-show- a.

Tracking

Tracking events are triggered automatically during ad rendering by showAds(). For client applications that perform their own ad rendering, the valid event types that must be handled are represented in the `tracking` array of the Ad Structure by:

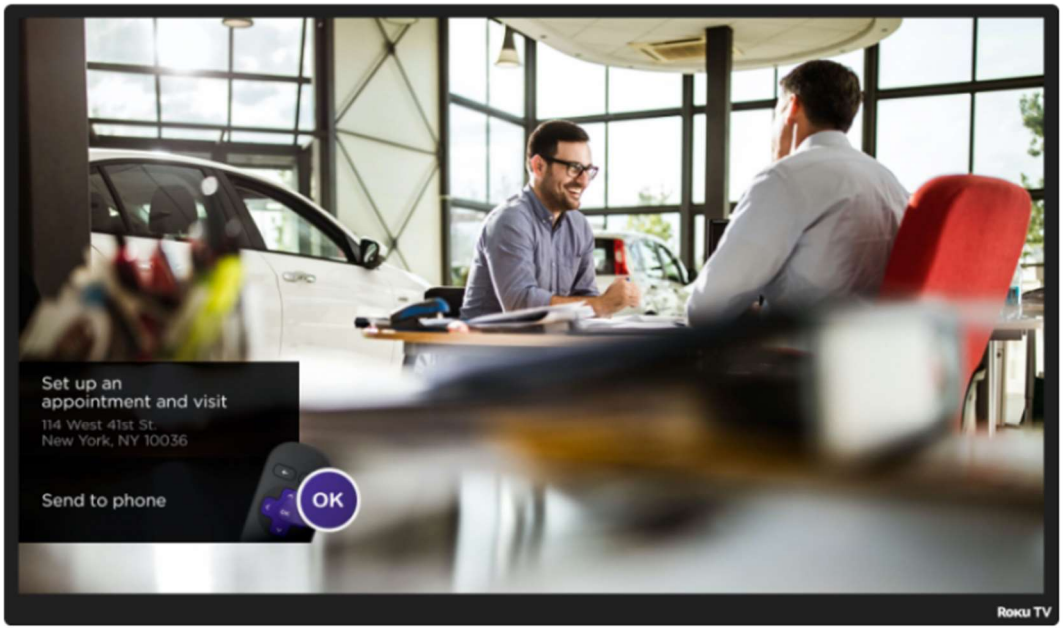
Event name	Trigger condition
Impression	Start of ad render (e.g., first frame of a video ad displayed)
PodStart	Beginning of ad pod render
PodComplete	Completed rendering ad pod
FirstQuartile	25% of video ad rendered
Midpoint	50% of video ad rendered
ThirdQuartile	75% of video ad rendered
Complete	100% of video ad rendered
Error	Error during ad parsing or rendering (VAST 3.0)
Close	User exited out of ad rendering before completion
Skip	User skipped ad (if skippable)
Pause	User paused ad
Resume	User resumed ad
Rewind	User rewound ad
Mute	User muted ad
Unmute	User un-muted ad
AcceptInvitation	User launched another portion of an ad (for interactive ads)

See e.g., https://developer.roku.com/docs/developer-program/advertising/integrating-roku-advertising-framework.md.

118. On information and belief, one or more components of the Roku system employ and provide a method for delivering an electronic message to an audience at an entertainment venue comprising the step of providing content on a first medium.

2. Direct-to-Store

A direct-to-store interactive ad shows an overlay with the physical address of a retail location near the streamer . By using the Roku remote, streamers can add the address to their phones.



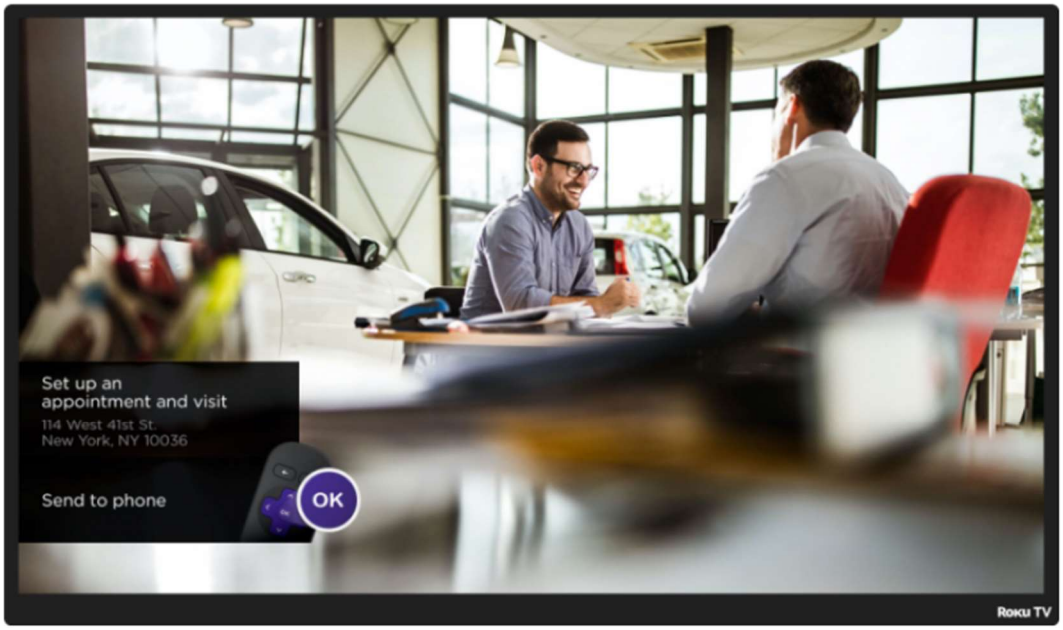
A direct-to-store interactive ad can help drive in-store traffic, show viewers important event information, or prompt them to set up appointments and visits.

See e.g., https://advertising.roku.com/resources/case-studies/roku-interactive-ads-helped-baskin-robbins-show-a.

119. On information and belief, one or more components of the Roku system employ and provide a method for delivering an electronic message to an audience at an entertainment venue comprising generating an electronic message based on the content and inserting additional auxiliary media content provided directly at the entertainment venue.

2. Direct-to-Store

A direct-to-store interactive ad shows an overlay with the physical address of a retail location near the streamer . By using the Roku remote, streamers can add the address to their phones.



A direct-to-store interactive ad can help drive in-store traffic, show viewers important event information, or prompt them to set up appointments and visits.

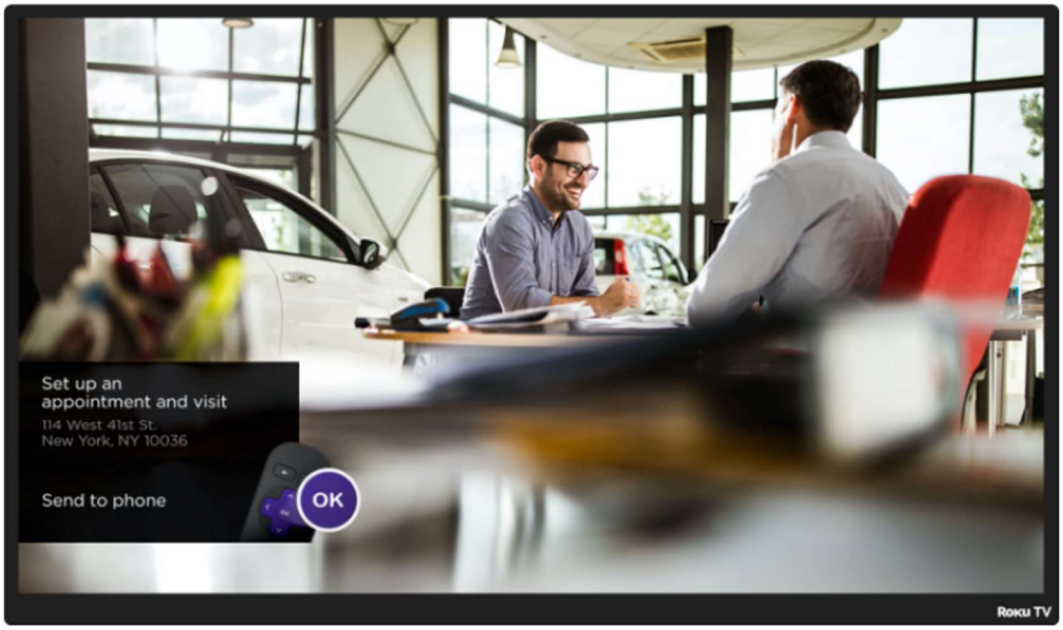
See e.g., https://advertising.roku.com/resources/case-studies/roku-interactive-ads-helped-baskin-robbins-show-a.

120. On information and belief, one or more components of the Roku system employ and provide a method that provides the electronic message based on the second content on a second medium to a subset of users based on a roster of devices in use for receiving the message that is synchronized with a portion of said first content based on

the instruction file.

2. Direct-to-Store

A direct-to-store interactive ad shows an overlay with the physical address of a retail location near the streamer . By using the Roku remote, streamers can add the address to their phones.



A direct-to-store interactive ad can help drive in-store traffic, show viewers important event information, or prompt them to set up appointments and visits.

See e.g., https://advertising.roku.com/resources/case-studies/roku-interactive-ads-helped-baskin-robbins-show-a.

121. Roku's direct infringement has damaged AdaptFlow and caused it to suffer and continue to suffer irreparable harm and damages.

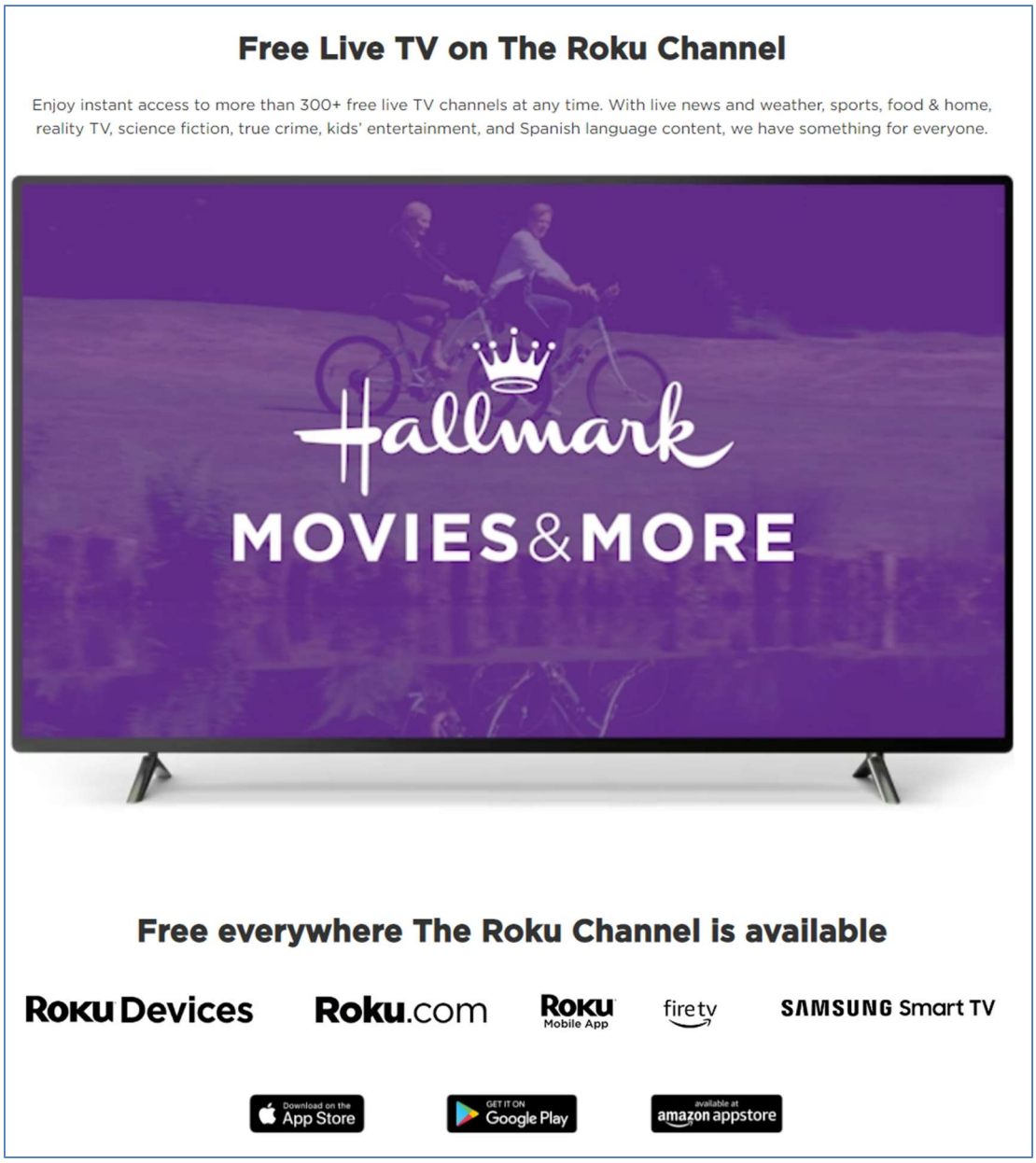
Count VII – Infringement of United States Patent No. 9,838,757

122. AdaptFlow repeats, realleges, and incorporates by reference, as if fully set forth here, the allegations of the preceding paragraphs above.

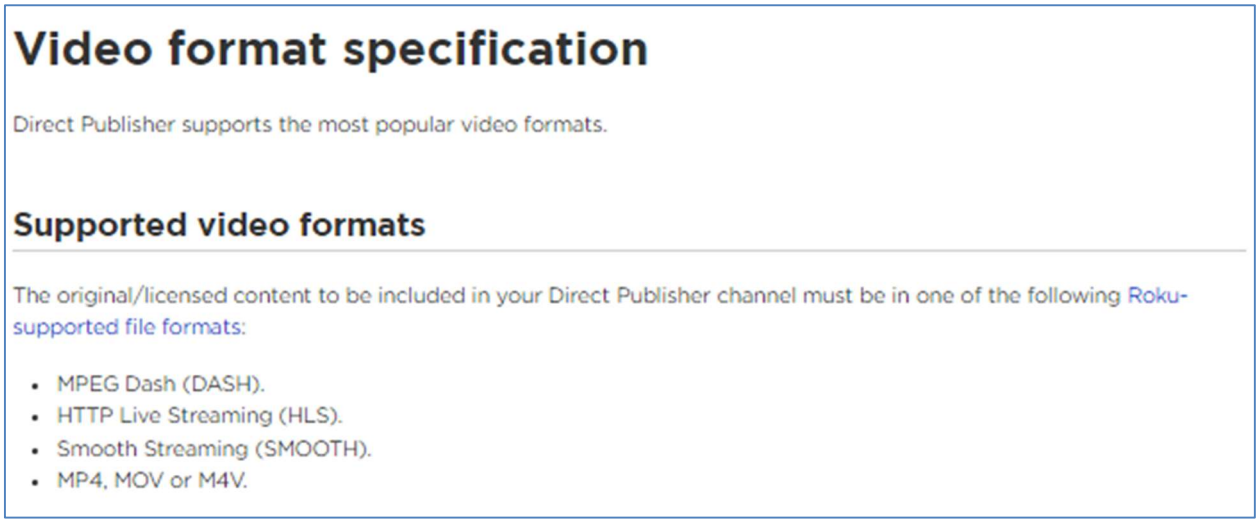
123. On information and belief, Roku (or those acting on its behalf) makes, uses, sells, imports and/or offers to sell the Roku Products and Services; and makes,

uses, sells, sells access to, imports, offers to sell and/or offers to sell access to the Roku System in the United States that infringe (literally and/or under the doctrine of equivalents) at least claim 1 of the '757 patent.

124. On information and belief, one or more components of the Roku system employ and provide a method comprising determining a structure of video data content to be transmitted.



See e.g., https://www.roku.com/whats-on/live-tv.



Video format specification

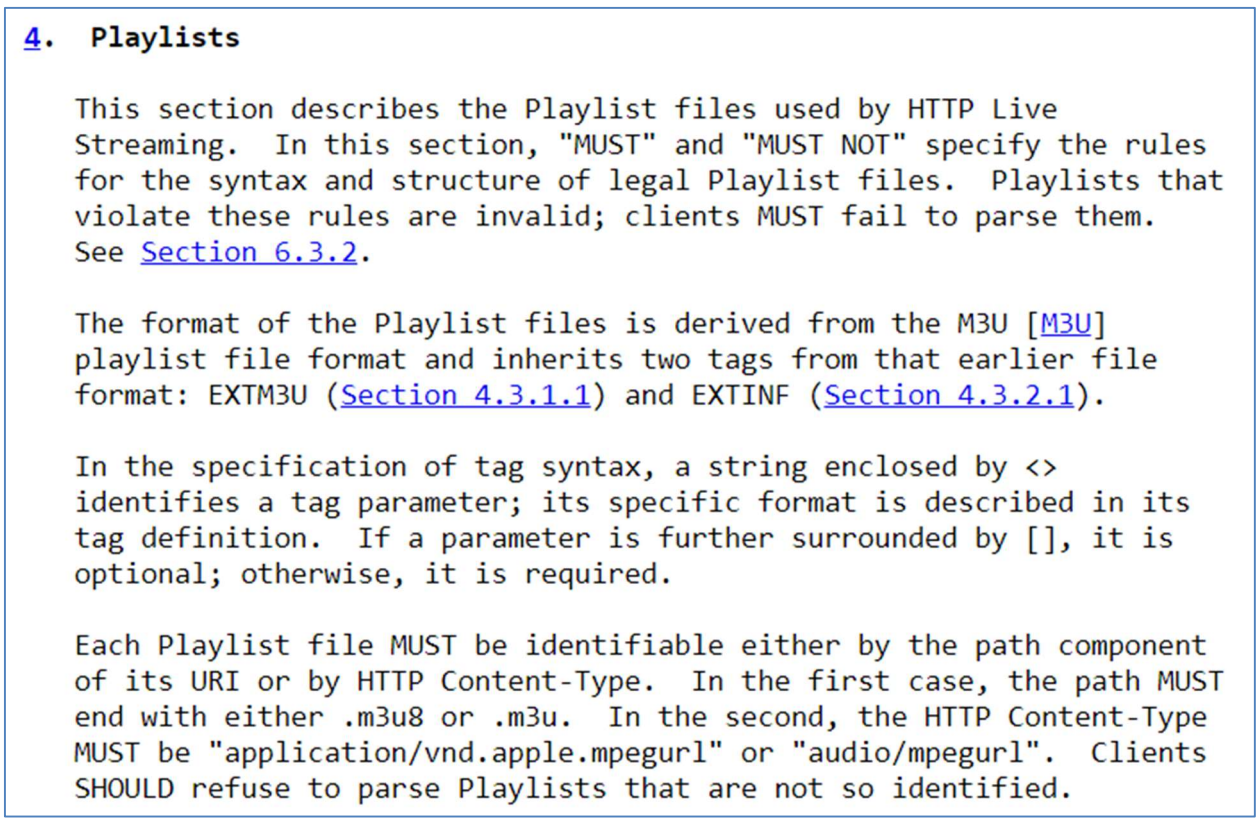
Direct Publisher supports the most popular video formats.

Supported video formats

The original/licensed content to be included in your Direct Publisher channel must be in one of the following Roku-supported file formats:

- MPEG Dash (DASH).
- HTTP Live Streaming (HLS).
- Smooth Streaming (SMOOTH).
- MP4, MOV or M4V.

See e.g., https://developer.roku.com/docs/direct-publisher/getting-started/content-specifications.md.



4. Playlists

This section describes the Playlist files used by HTTP Live Streaming. In this section, "MUST" and "MUST NOT" specify the rules for the syntax and structure of legal Playlist files. Playlists that violate these rules are invalid; clients MUST fail to parse them. See Section 6.3.2.

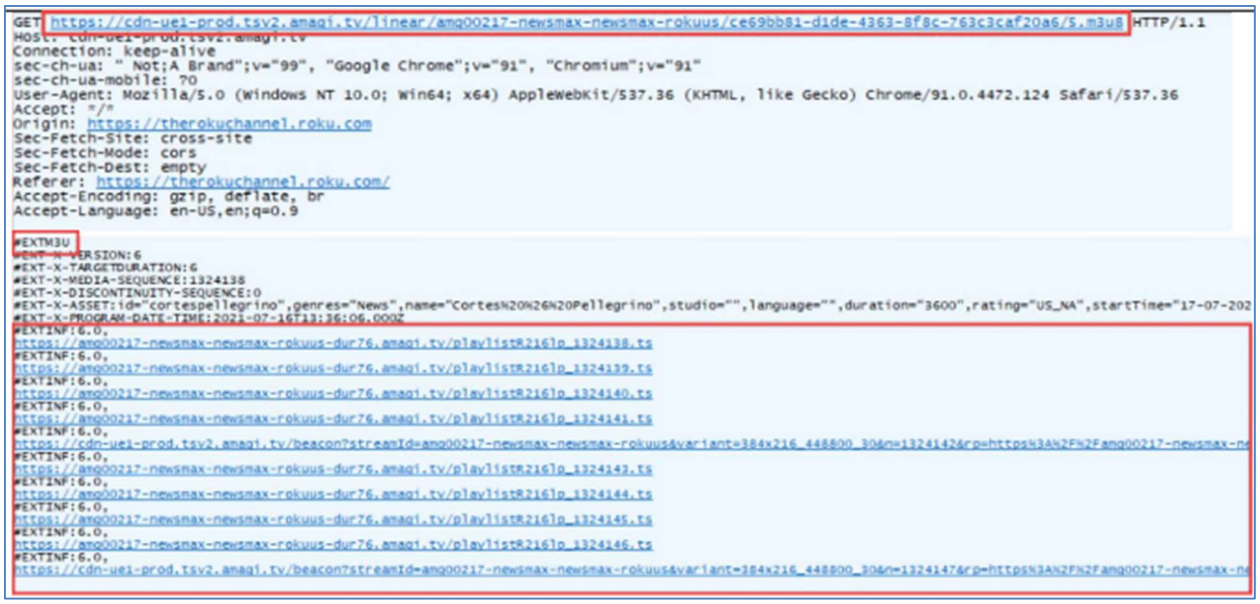
The format of the Playlist files is derived from the M3U [M3U] playlist file format and inherits two tags from that earlier file format: EXTM3U (Section 4.3.1.1) and EXTINF (Section 4.3.2.1).

In the specification of tag syntax, a string enclosed by <> identifies a tag parameter; its specific format is described in its tag definition. If a parameter is further surrounded by [], it is optional; otherwise, it is required.

Each Playlist file MUST be identifiable either by the path component of its URI or by HTTP Content-Type. In the first case, the path MUST end with either .m3u8 or .m3u. In the second, the HTTP Content-Type MUST be "application/vnd.apple.mpegurl" or "audio/mpegurl". Clients SHOULD refuse to parse Playlists that are not so identified.

See e.g., https://datatracker.ietf.org/doc/html/rfc8216.

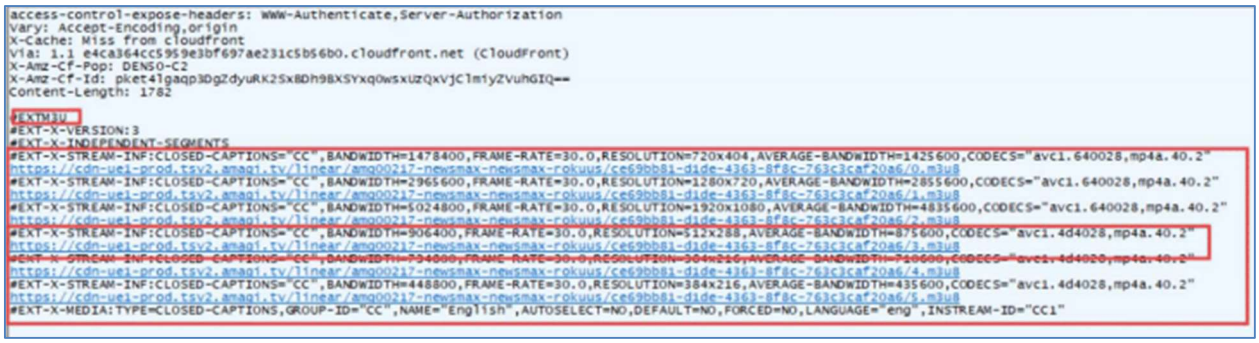
See e.g., Fiddler capture of a Roku live program.



See e.g., Fiddler capture of a Roku live program.

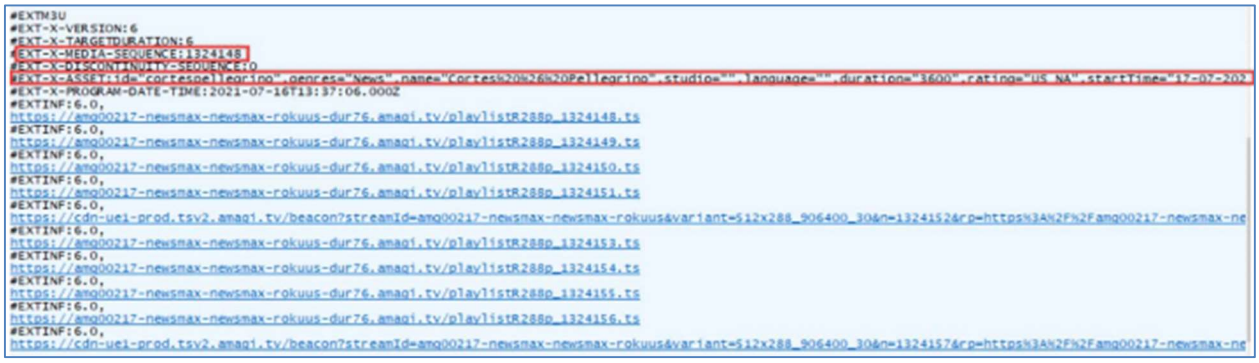
125. On information and belief, one or more components of the Roku system employ and provide a method comprising the splitting the structure of video data into chunks of a hierarchical manner, wherein each said chunk comprises one or more nodes representing metadata for a respective portion of video data content, a first chunk

comprises a description of one or more nodes included in the first chunk and a second chunk comprises information indicating that a second description of one or more nodes included in the second chunk is included in the next chunk, the first and second descriptions each comprise of one or more node identifiers, a name representing video information with one or more associated node, and a timecode indicating the duration of the video information.

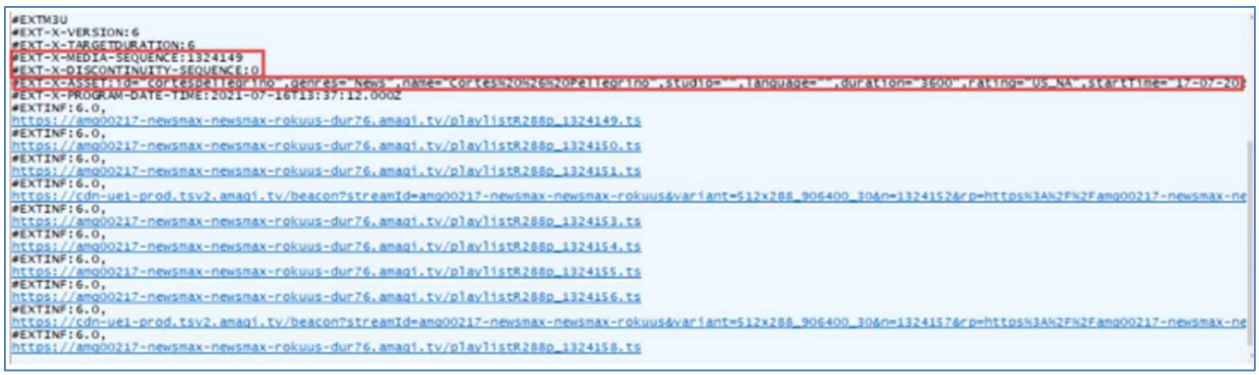


See e.g., Fiddler capture of a Roku live program.

See e.g., Fiddler capture of a Roku live program.



See e.g., Fiddler capture of a Roku live program.



See e.g., Fiddler capture of a Roku live program.

See e.g., Fiddler capture of a Roku live program.

4.3.2.3. EXT-X-DISCONTINUITY

The EXT-X-DISCONTINUITY tag indicates a discontinuity between the Media Segment that follows it and the one that preceded it.

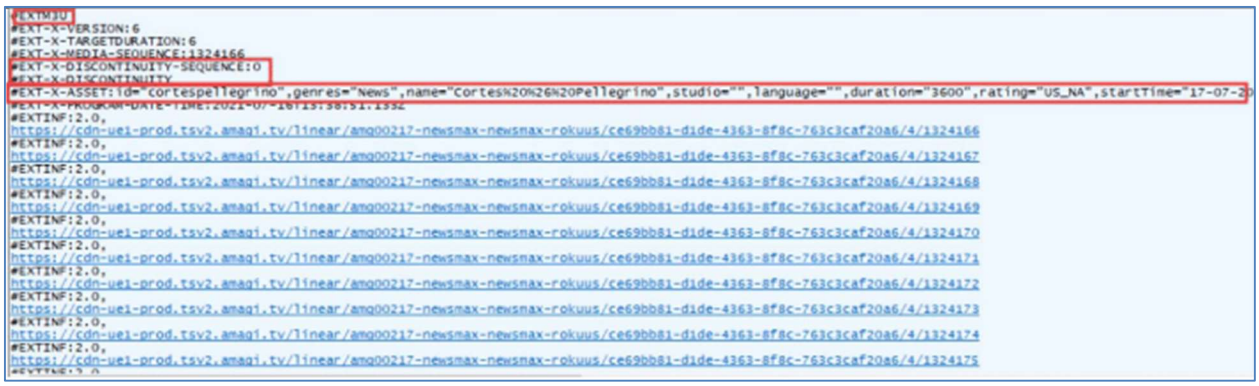
Its format is:

#EXT-X-DISCONTINUITY

The EXT-X-DISCONTINUITY tag MUST be present if there is a change in any of the following characteristics:

- file format
- number, type, and identifiers of tracks
- timestamp sequence

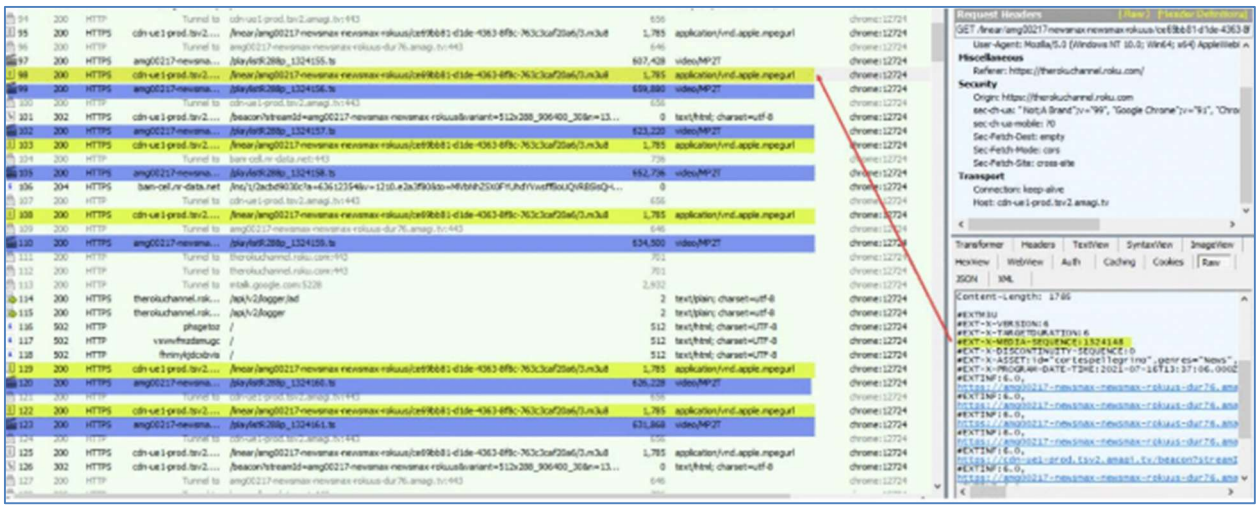
See e.g., https://datatracker.ietf.org/doc/html/rfc8216.



See e.g., Fiddler capture of a Roku live program.

See e.g., Fiddler capture of a Roku live program.

126. On information and belief, one or more components of the Roku system employ and provide a method comprising transmitting chunks of data in a data stream at successive time intervals so that each chunk is transmitted before said respective portion of said video data content that the respective chunk is related to.



See e.g., Fiddler capture of a Roku live program.

127. Roku's direct infringement has damaged AdaptFlow and caused it to suffer and continue to suffer irreparable harm and damages

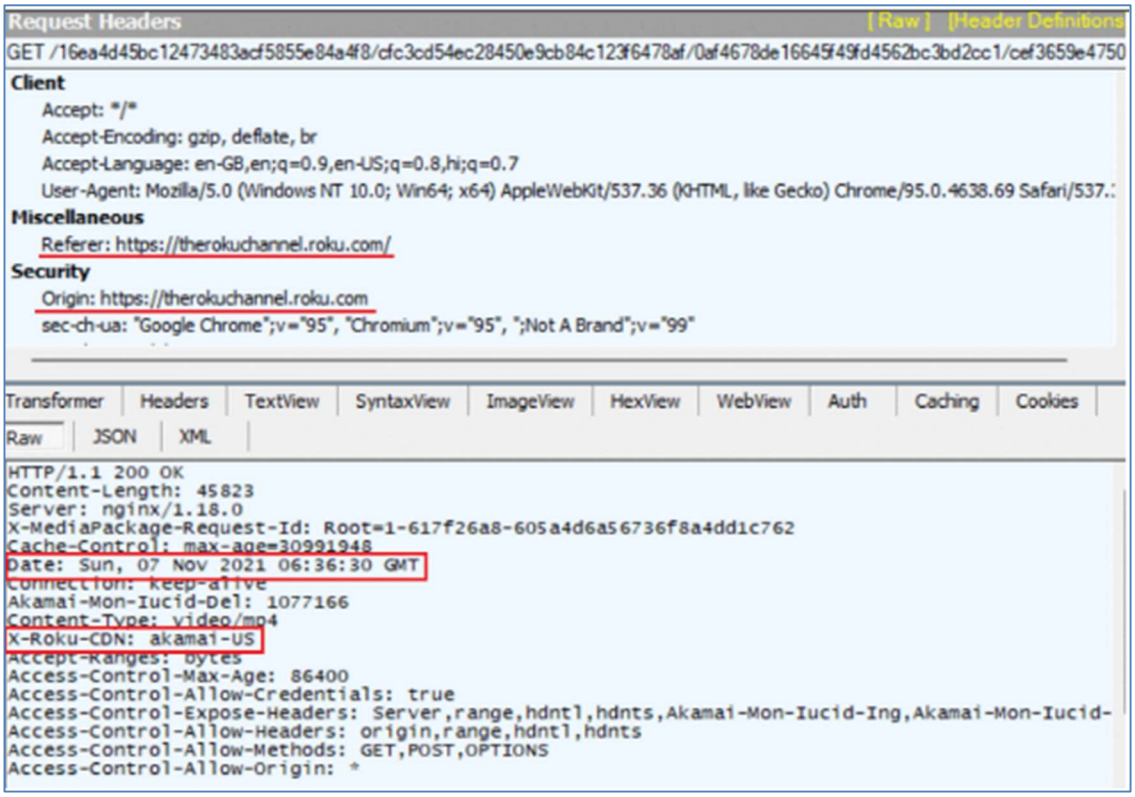
Count VIII – Infringement of United States Patent No. 10,015,064

128. AdaptFlow repeats, realleges, and incorporates by reference, as if fully set forth here, the allegations of the preceding paragraphs above.

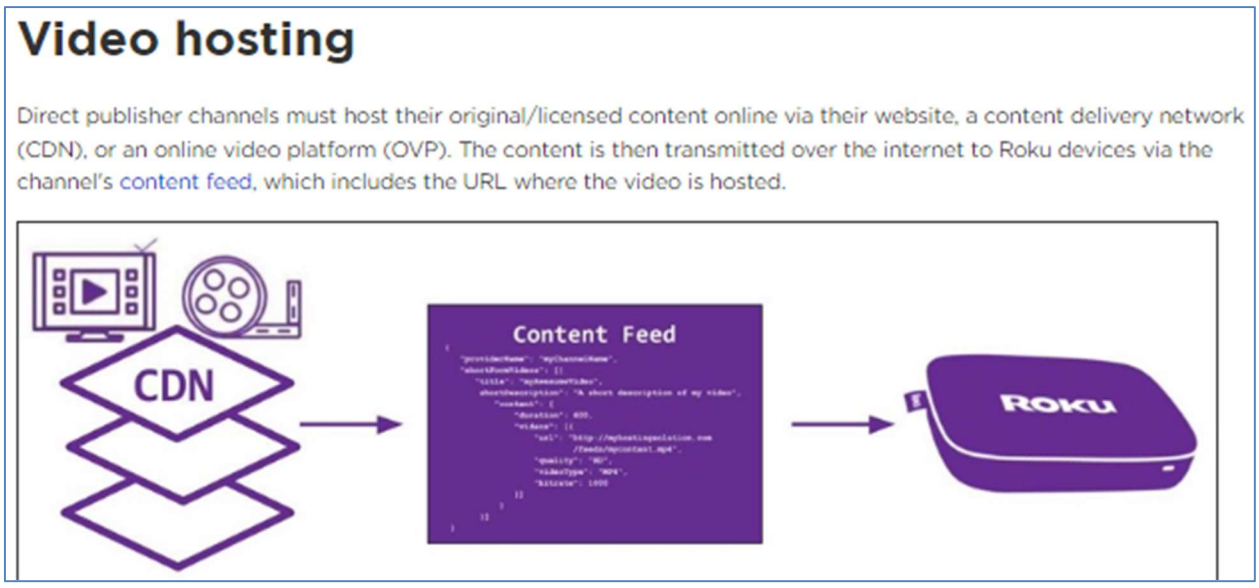
129. On information and belief, Roku (or those acting on its behalf) makes, uses, sells, imports and/or offers to sell the Roku Products and Services; and makes, uses, sells, sells access to, imports, offers to sell and/or offers to sell access to the Roku System in the United States that infringe (literally and/or under the doctrine of equivalents) at least claim 1 of the '064 patent.

130. On information and belief, one or more components of the Roku system employ and provide a method comprising the monitoring of data traffic at a gateway coupling between first and second data networks, the first data network comprises of a plurality of devices for accessing content, the devices have an input for receiving user

selection of contents and a second network comprising a plurality of content servers to deliver a plurality of contents. The monitoring collects control data and associating time stamps to the collected data, which defines where and how to obtain the content.



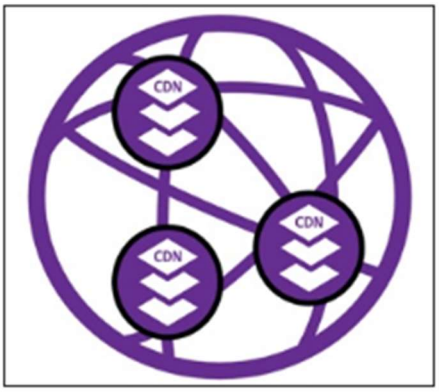
See e.g., Fiddler capture of Roku live player.



See e.g., https://developer.roku.com/docs/direct-publisher/getting-started/video-hosting.md.

Content Delivery Networks (CDNs)

Content is typically hosted on a CDN, which is a group of servers located around the world. Using a CDN ensures that your content can be streamlined to many users at different locations at the same time without any bottlenecks.



Direct publisher channels can use any CDN that is able to stream content. The following list includes some of the CDNs that publishers have used to host content for their Roku channels:

- AdvergentCDN
- Akamai
- Amazon Web Services (AWS)
- BitGravity
- Boxcast
- Brightcove
- Comcast Technology Solutions
- EdgeCast
- Level 3
- Limelight Networks
- Scale Engine

See e.g., https://developer.roku.com/docs/direct-publisher/getting-started/video-hosting.md.

This Privacy Policy applies to information collected by Roku through the **"Roku Services"**, which includes: Roku websites (**"Roku Sites"**), Roku's products and certain other products utilizing Roku's streaming platform or operating systems (including Roku players, Roku TVs, speakers and other audio-enabled products) (collectively, **"Roku Devices"**), certain Roku co-branded channels (**"Roku Direct Publisher Channels"**), Roku's mobile apps (including the Roku Mobile App), Roku branded channels like the Roku Media Player and The Roku Channel (**"Roku's Channels"**), Roku's advertising services, the Roku emails and newsletters, and Roku's accounts on third-party social networking sites.

Using our products and services, you can stream channels provided by other companies (**"Third-Party Channels"**), and Roku co-branded channels like the Roku Direct Publisher Channels. When you install, access or stream one of these channels, you are also interacting with that channel directly, and the channel provider and its analytics and advertising partners may collect information from your Roku Device, including your device identifier and how you use that channel. This Privacy Policy does not cover the actions of these channel providers and their partners. For more information about how these channels use your data, please see their privacy policies.

Part I. Information Collection

Information is collected from or about you from various sources, as described below.

A. Information You Provide

1. Registration Information

When you sign up for a Roku account, we ask for information such as your name, email address, postal address, telephone number, birth date, and demographic information. If you sign up using a social media account, we may also receive information from those social networking services, for example, your name or username. We also may collect similar contact information if you register to receive products or services, or copies of Roku blog posts.

* * *

B. Information We Collect Automatically Through the Roku Services

1. Device Information

We may receive information about the browser and devices you use to access the Internet, including our services, such as device types and models, unique identifiers (including, for Roku Devices, the Advertising Identifier associated with that device), MAC address, IP address, operating system type and version, browser type and language, Wi-Fi network name and connection data, and information about other devices connected to the same network. We may also gather the WiFi MAC addresses and broadcast signal strengths of your router and other Wi-Fi routers in your area. For Roku Devices, we may also collect the name of the retailer to whom your Roku Device was shipped, various quality measures, error logs, software version numbers, and device status (including the status of battery-powered accessories). When you enable Bluetooth on a Roku Device, we may collect your Bluetooth usage, such as connection quality, the name of the device connected to your Roku Device, and the start and stop time of your connection.

We may infer your general location information and internet service provider from the device information we collect (for example, IP address and MAC address may be used to infer your general geographic area).

* * *

https://www.akamai.com/our-thinking/cdn/what-is-a-cdn

3. Activity, Location, and Usage Information on Roku Services

We receive information about your interactions with the Roku Services, such as your browsing history, search results, audio information when you use voice-enabled features, interactions with content and ads, and settings and preferences. When you access channels on a Roku device or Roku's Channels on other devices, we may receive information about your activities, like the channels you access (including usage statistics such as what channels you access, the time you access them, and how long you spend viewing them), and information about the videos and other content you select and view within these channels. If you use the Roku Media Player channel to view your video or photo files or listen to your music files, Roku will collect data about the files viewed within the Roku Media Player, such as codecs, and other metadata of the local files you play through the Roku Media Player.When you use a Roku TV with the Smart TV experience enabled, we use Automatic Content Recognition (**"ACR"**) technology to receive information about what you watch via the Roku TV's antenna (including live television content and ads), and via devices connected to your Roku TV (including streaming players, consoles and cable and satellite set top boxes). For example, we receive TV viewing information such as the programs, commercials, and channels you view, the date, time and duration of the viewing, and how you use the on-screen TV guide. We receive TV viewing information both when you access live TV directly through your Smart TV's interface and when you access live TV from within a Third-Party Channel. If the Smart TV experience is enabled on your Roku TV, we will use this information to personalize your TV viewing experience and ads.

If you use Roku's mobile apps, in addition to the other usage information described in this section, we also log whether you use a feature called Play on Roku, which allows you to play videos, photos and music stored on your mobile device on your Roku TV or a TV with a Roku streaming player connected to it. Roku's mobile apps may need permissions to access the content and other information stored on your mobile device.

Through Roku's advertising services, we receive information about when, where and how an ad was displayed, the content you were viewing when the ad was served, and your clicks and any other information you provide when you interact with the ad (such as your phone number in order to receive a coupon code or your zip code to find an advertiser's closest locations). For example, when Roku provides advertising services in Third-Party Channels, Roku receives information related to the ad, such as the name of the Channel and the genre, language, and rating of the content you are watching when Roku serves the ad.

When you visit websites, apps, channels, and connected devices (including Smart TVs) to which Roku provides advertising or measurement and analytics services, we may receive information about your activities, including the content you view, the date and time of your visits, how you interact with these websites, apps and devices, and how you interact and respond to ads. We may also receive your precise geolocation information and use it for advertising purposes, including ad targeting.

See e.g., https://docs.roku.com/published/userprivacypolicy/en/us.

What is a CDN (Content Delivery Network)?

A content delivery network (CDN) is a group of geographically distributed servers that speed up the delivery of web content by bringing it closer to where users are. Data centers across the globe use caching, a process that temporarily stores copies of files, so that you can access internet content from a web-enabled device or browser more quickly through a server near you. CDNs cache content like web pages, images, and video in proxy servers near to your physical location. This allows you to do things like watch a movie, download software, check your bank balance, post on social media, or make purchases, without having to wait for content to load.

What is an example of a CDN?

A large portion of all internet content is delivered through CDNs. Here is a simple example:

If you were in New York and wanted to view the website of your favorite store in London that's hosted on a server in the UK, you would experience slow content load times if the request had to travel all the way across the Atlantic Ocean. To remedy this, a CDN would store a cached version of the London website content in multiple geographical locations around the world, also called "points of presence" (PoPs). These PoPs contain their own caching servers and are responsible for delivering that content close to where you're located in New York.

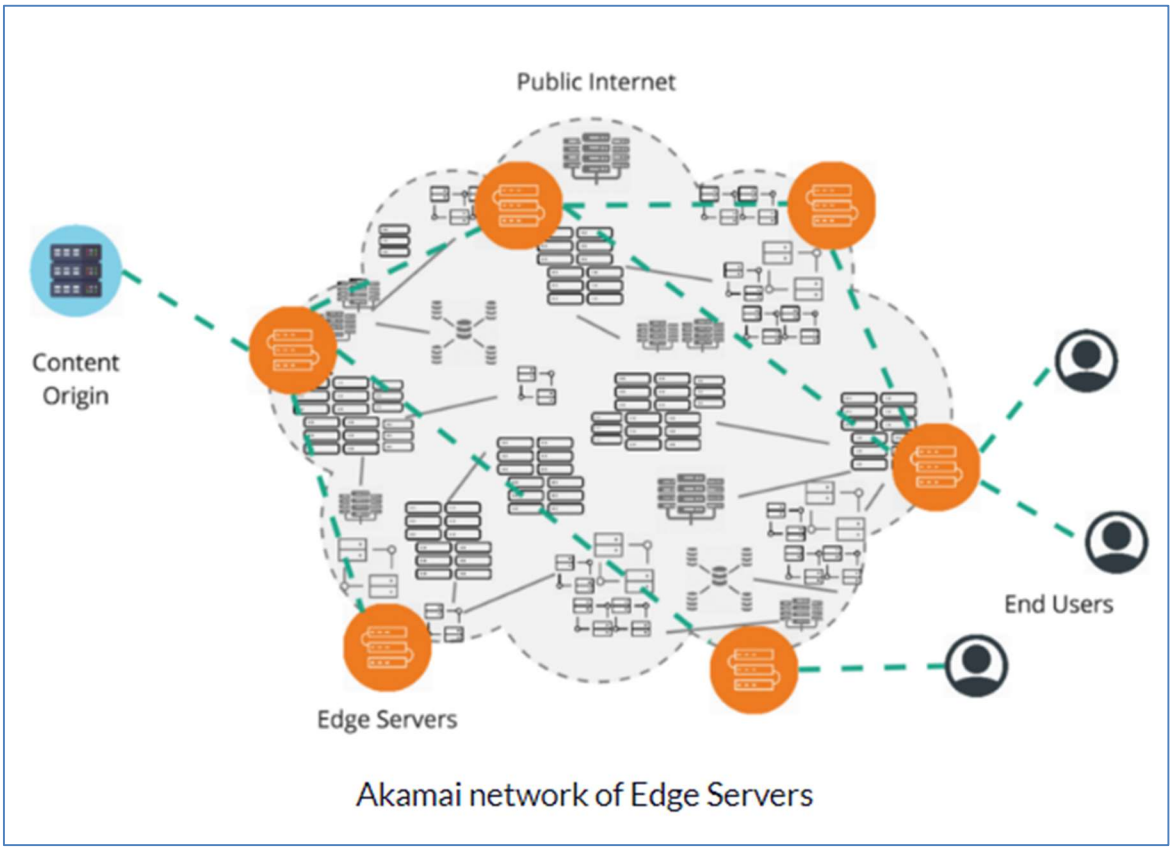
Content delivered from a server closest to your physical location gives you a faster, high-performance web experience.

See e.g., https://www.akamai.com/our-thinking/cdn/what-is-a-cdn.

Key Features

1 **IP Anycast**: Provide a decentralized DNS service to users that enables the creation of a logical name server comprising multiple physical servers deployed across multiple networks and continents.

See e.g., https://developer.akamai.com/edge-dns.



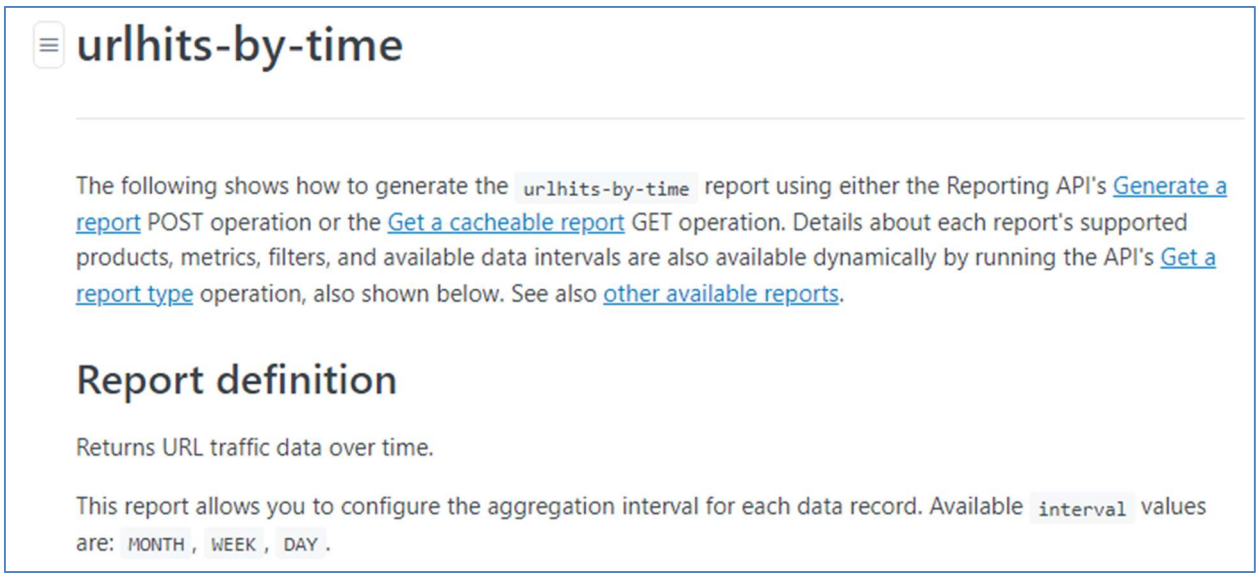
When a user makes a request, Akamai dynamically maps this to the closest available edge server. The edge server applies the business rules that the content provider has specified, before using the best available route between all other edge servers within the Akamai network to fetch content from the origin. Business rules are replicated on each edge server.

Any content available and configured to be cached is then cached on the edge server for future requests connecting to that node. We will look at this in more detail later on.

Akamai provides a web GUI named the 'Luna Control Center', several APIs, and a CLI.

As seen in the *Monitor* tab, many reporting and analytics tools are available for generating insights at a CDN level. Logs from edge servers are also available on request.

See e.g., https://www.freecodecamp.org/news/an-introduction-to-the-akamai-content-delivery-network-806aa16d8781/.



urlhits-by-time

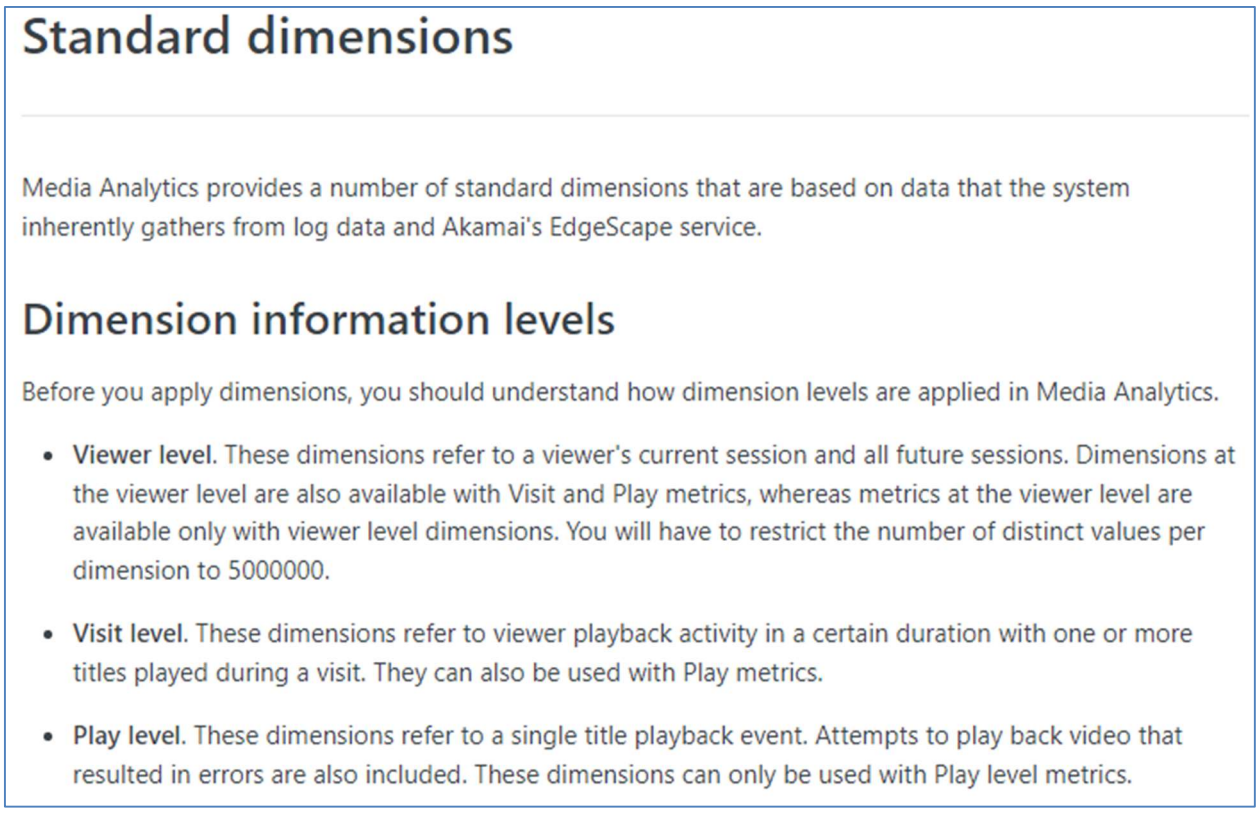
The following shows how to generate the `urlhits-by-time` report using either the Reporting API's Generate a report POST operation or the Get a cacheable report GET operation. Details about each report's supported products, metrics, filters, and available data intervals are also available dynamically by running the API's Get a report type operation, also shown below. See also other available reports.

Report definition

Returns URL traffic data over time.

This report allows you to configure the aggregation interval for each data record. Available `interval` values are: `MONTH`, `WEEK`, `DAY`.

See e.g., https://techdocs.akamai.com/reporting/reference/urlhits-by-time.



Standard dimensions

Media Analytics provides a number of standard dimensions that are based on data that the system inherently gathers from log data and Akamai's EdgeScape service.

Dimension information levels

Before you apply dimensions, you should understand how dimension levels are applied in Media Analytics.

- **Viewer level**. These dimensions refer to a viewer's current session and all future sessions. Dimensions at the viewer level are also available with Visit and Play metrics, whereas metrics at the viewer level are available only with viewer level dimensions. You will have to restrict the number of distinct values per dimension to 5000000.
- **Visit level**. These dimensions refer to viewer playback activity in a certain duration with one or more titles played during a visit. They can also be used with Play metrics.
- **Play level**. These dimensions refer to a single title playback event. Attempts to play back video that resulted in errors are also included. These dimensions can only be used with Play level metrics.

The dimensions

The following standard dimensions are supported:

Dimension	Definition
ASNUM	Autonomous System Number. This identifies the internet network.
CDN	Content Distribution Network. This identifies the CDN used to deliver the content. For example, <>.
Connection Speed	The internet connection speed of the viewer. Available values are: • 1 to 55 Kbps: BW = 1 • 56 to 255 Kbps: BW = 56 • 256 to 999 Kbps: BW = 256 • 1000 to 1999 Kbps: BW = 1000 • 2000 to 4999 Kbps: BW = 2000 • 5000 Kbps+ : BW = 5000
Connection Type	The internet connection type. For example, Wi-Fi or cellular.
DMA	Designated Market Area.
Day of week	Day as per the time zone specified in the Analyzer.

Event Name	Name of the event.
Event URL	Stream URL.
Format	The format in which the media is delivered. Available values are: • WMS • Flash • Progressive Download
Geography	Viewer location. Available granularities are: • Continent • Region • Country • City (available for the US and Canada only)
Time	Time as per the time zone specified in the Analyzer.
Referer	URL of the webpage that refers to the requested media.

See e.g., https://techdocs.akamai.com/media-analytics/docs/standard-dimensions.

131. On information and belief, one or more components of the Roku system employ and provide a method comprising the analyzing of data traffic to identify a group of contents that are more-frequently-accessed by the plurality of devices, where data is collected and time-stamped and correlated in accordance with the time of day, and day of the week at which each one of the content servers was accessed by a user to determine an access pattern for each of said more-frequently- accessed contents, wherein collected time-stamped data are inferred from previous user selections.

This Privacy Policy applies to information collected by Roku through the **"Roku Services"**, which includes: Roku websites (**"Roku Sites"**), Roku's products and certain other products utilizing Roku's streaming platform or operating systems (including Roku players, Roku TVs, speakers and other audio-enabled products) (collectively, **"Roku Devices"**), certain Roku co-branded channels (**"Roku Direct Publisher Channels"**), Roku's mobile apps (including the Roku Mobile App), Roku branded channels like the Roku Media Player and The Roku Channel (**"Roku's Channels"**), Roku's advertising services, the Roku emails and newsletters, and Roku's accounts on third-party social networking sites.

Using our products and services, you can stream channels provided by other companies (**"Third-Party Channels"**), and Roku co-branded channels like the Roku Direct Publisher Channels. When you install, access or stream one of these channels, you are also interacting with that channel directly, and the channel provider and its analytics and advertising partners may collect information from your Roku Device, including your device identifier and how you use that channel. This Privacy Policy does not cover the actions of these channel providers and their partners. For more information about how these channels use your data, please see their privacy policies.

Part I. Information Collection

Information is collected from or about you from various sources, as described below.

A. Information You Provide

1. Registration Information

When you sign up for a Roku account, we ask for information such as your name, email address, postal address, telephone number, birth date, and demographic information. If you sign up using a social media account, we may also receive information from those social networking services, for example, your name or username. We also may collect similar contact information if you register to receive products or services, or copies of Roku blog posts.

* * *

B. Information We Collect Automatically Through the Roku Services

1. Device Information

We may receive information about the browser and devices you use to access the Internet, including our services, such as device types and models, unique identifiers (including, for Roku Devices, the Advertising Identifier associated with that device), MAC address, IP address, operating system type and version, browser type and language, Wi-Fi network name and connection data, and information about other devices connected to the same network. We may also gather the WiFi MAC addresses and broadcast signal strengths of your router and other Wi-Fi routers in your area. For Roku Devices, we may also collect the name of the retailer to whom your Roku Device was shipped, various quality measures, error logs, software version numbers, and device status (including the status of battery-powered accessories). When you enable Bluetooth on a Roku Device, we may collect your Bluetooth usage, such as connection quality, the name of the device connected to your Roku Device, and the start and stop time of your connection.

We may infer your general location information and internet service provider from the device information we collect (for example, IP address and MAC address may be used to infer your general geographic area).

* * *

3. Activity, Location, and Usage Information on Roku Services

We receive information about your interactions with the Roku Services, such as your browsing history, search results, audio information when you use voice-enabled features, interactions with content and ads, and settings and preferences. When you access channels on a Roku device or Roku's Channels on other devices, we may receive information about your activities, like the channels you access (including usage statistics such as what channels you access, the time you access them, and how long you spend viewing them), and information about the videos and other content you select and view within these channels. If you use the Roku Media Player channel to view your video or photo files or listen to your music files, Roku will collect data about the files viewed within the Roku Media Player, such as codecs, and other metadata of the local files you play through the Roku Media Player.When you use a Roku TV with the Smart TV experience enabled, we use Automatic Content Recognition (**"ACR"**) technology to receive information about what you watch via the Roku TV's antenna (including live television content and ads), and via devices connected to your Roku TV (including streaming players, consoles and cable and satellite set top boxes). For example, we receive TV viewing information such as the programs, commercials, and channels you view, the date, time and duration of the viewing, and how you use the on-screen TV guide. We receive TV viewing information both when you access live TV directly through your Smart TV's interface and when you access live TV from within a Third-Party Channel. If the Smart TV experience is enabled on your Roku TV, we will use this information to personalize your TV viewing experience and ads.

If you use Roku's mobile apps, in addition to the other usage information described in this section, we also log whether you use a feature called Play on Roku, which allows you to play videos, photos and music stored on your mobile device on your Roku TV or a TV with a Roku streaming player connected to it. Roku's mobile apps may need permissions to access the content and other information stored on your mobile device.

Through Roku's advertising services, we receive information about when, where and how an ad was displayed, the content you were viewing when the ad was served, and your clicks and any other information you provide when you interact with the ad (such as your phone number in order to receive a coupon code or your zip code to find an advertiser's closest locations). For example, when Roku provides advertising services in Third-Party Channels, Roku receives information related to the ad, such as the name of the Channel and the genre, language, and rating of the content you are watching when Roku serves the ad.

When you visit websites, apps, channels, and connected devices (including Smart TVs) to which Roku provides advertising or measurement and analytics services, we may receive information about your activities, including the content you view, the date and time of your visits, how you interact with these websites, apps and devices, and how you interact and respond to ads. We may also receive your precise geolocation information and use it for advertising purposes, including ad targeting.

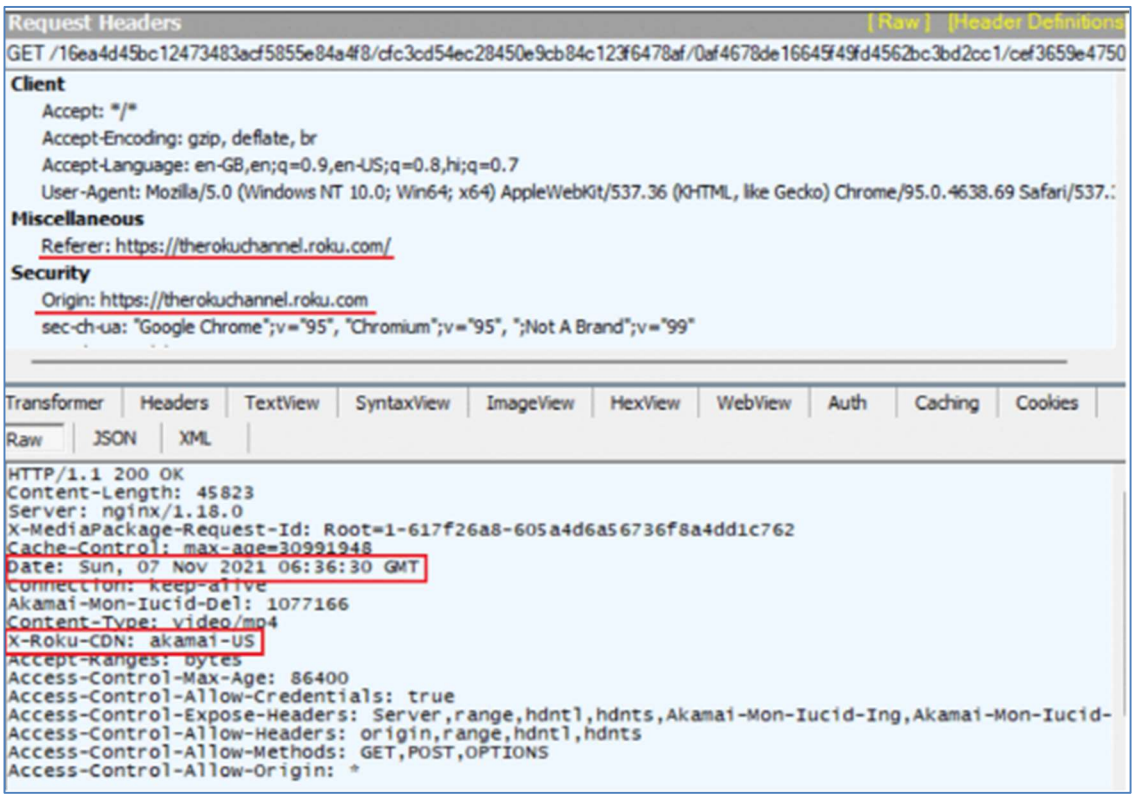
See e.g., https://docs.roku.com/published/userprivacypolicy/en/us.

How does a CDN work?

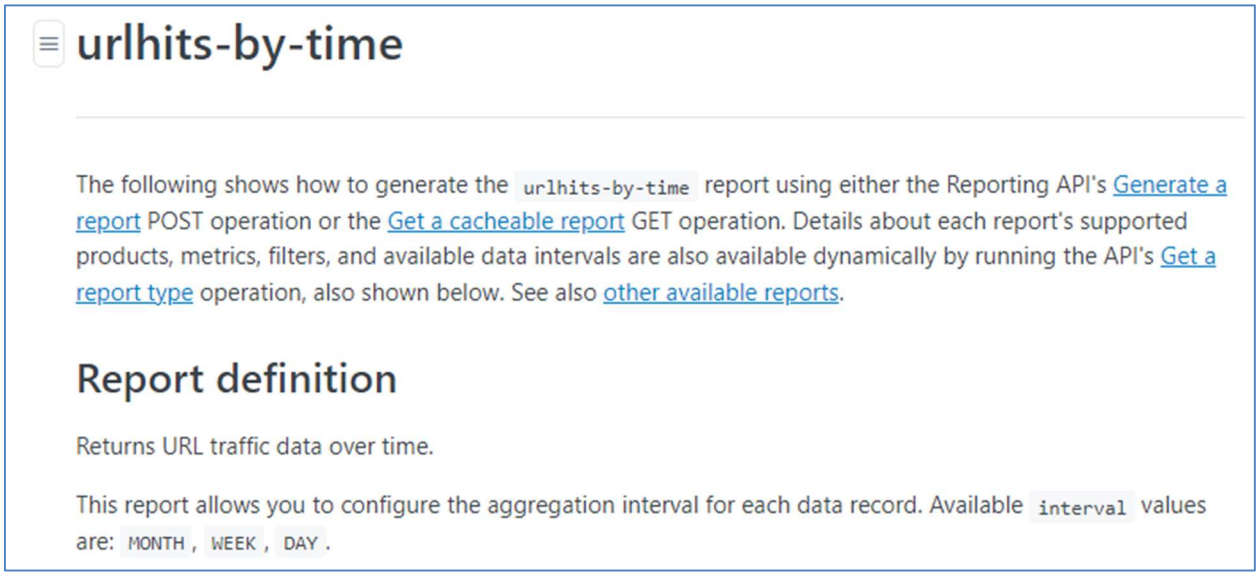
The mission of a CDN is to reduce *latency.* Latency is that annoying delay you experience when trying to access a web page or video stream before it fully loads on your device. Although measured in milliseconds, it can feel like forever, and may even result in a load error or time-out. Some content delivery networks alleviate latency by reducing the physical distance that the content needs to travel to reach you. Therefore, larger, more widely distributed CDNs are able to deliver web content more quickly and reliably by putting the content as close to the end user as possible.

Let's say it's the weekend and you want to kick back and stream the latest Hollywood movie release — the CDN finds an optimal server on its network to serve up that video. Usually, that will be the server closest to your physical location. The media files will be cached and remain on that content delivery network server for other user requests in the same geographic area. If the content you requested is unavailable or outdated, the CDN service will store the newly fetched content to serve any future requests.

See e.g., https://www.akamai.com/our-thinking/cdn/what-is-a-cdn.



See e.g., Fiddler capture of Roku live player.



See e.g., https://techdocs.akamai.com/reporting/reference/urlhits-by-time.

Standard dimensions

Media Analytics provides a number of standard dimensions that are based on data that the system inherently gathers from log data and Akamai's EdgeScape service.

Dimension information levels

Before you apply dimensions, you should understand how dimension levels are applied in Media Analytics.

- **Viewer level**. These dimensions refer to a viewer's current session and all future sessions. Dimensions at the viewer level are also available with Visit and Play metrics, whereas metrics at the viewer level are available only with viewer level dimensions. You will have to restrict the number of distinct values per dimension to 5000000.
- **Visit level**. These dimensions refer to viewer playback activity in a certain duration with one or more titles played during a visit. They can also be used with Play metrics.
- **Play level**. These dimensions refer to a single title playback event. Attempts to play back video that resulted in errors are also included. These dimensions can only be used with Play level metrics.

The dimensions

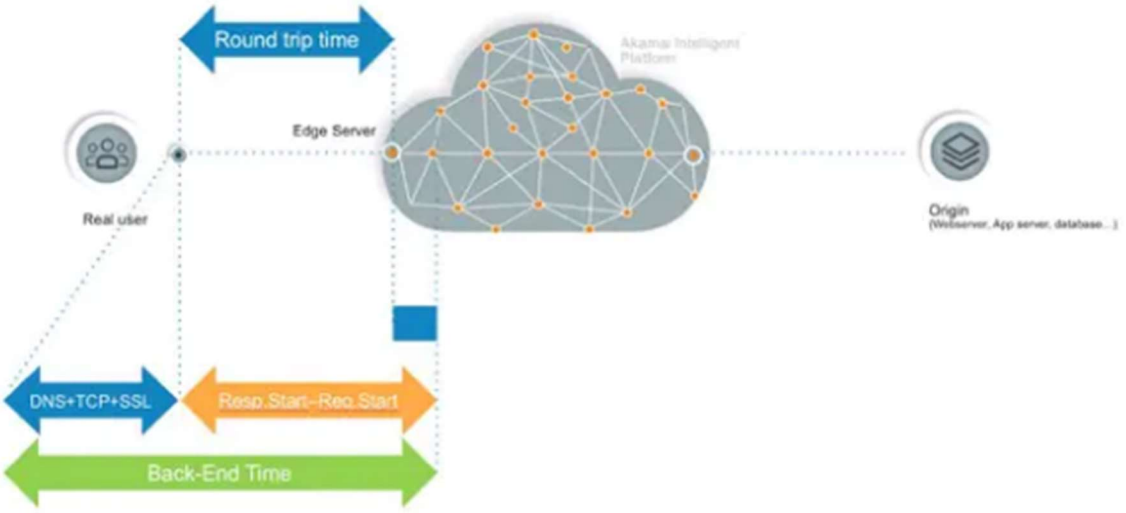
The following standard dimensions are supported:

Dimension	Definition
ASNUM	Autonomous System Number. This identifies the internet network.
CDN	Content Distribution Network. This identifies the CDN used to deliver the content. For example, <>.
Connection Speed	The internet connection speed of the viewer. Available values are: • 1 to 55 Kbps: BW = 1 • 56 to 255 Kbps: BW = 56 • 256 to 999 Kbps: BW = 256 • 1000 to 1999 Kbps: BW = 1000 • 2000 to 4999 Kbps: BW = 2000 • 5000 Kbps+ : BW = 5000
Connection Type	The internet connection type. For example, Wi-Fi or cellular.
DMA	Designated Market Area.
Day of week	Day as per the time zone specified in the Analyzer.

Event Name	Name of the event.
Event URL	Stream URL.
Format	The format in which the media is delivered. Available values are: • WMS • Flash • Progressive Download
Geography	Viewer location. Available granularities are: • Continent • Region • Country • City (available for the US and Canada only)
Time	Time as per the time zone specified in the Analyzer.
Referer	URL of the webpage that refers to the requested media.

See e.g., https://techdocs.akamai.com/media-analytics/docs/standard-dimensions.

2. **Cache hit example**: When there is a cache hit (i.e., content is served by an edge server), the request will *not* go all the way back to the origin, as shown in the diagram below.



See e.g., https://developer.akamai.com/blog/2018/11/16/using-cdn-server-timing-monitor-cdn-and-origin-performance.

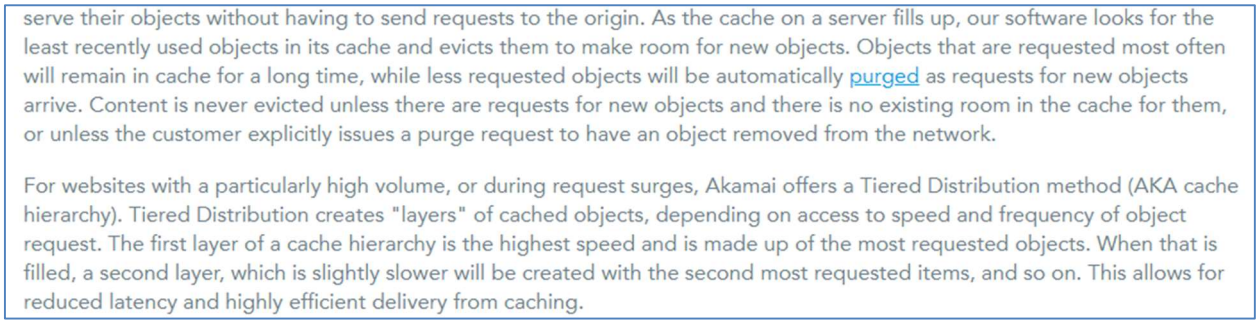
132. On information and belief, one or more components of the Roku system employ and provide a method comprising pre-fetching the more-frequently-accessed contents according to their control data and collected time-stamped data.

Prefetching

Learn

When a user requests a media asset, especially large media assets, they may only be asking for a portion of a stream. The HTTP chunked media streaming formats, such as Akamai HD Flash, Adobe HDS, Apple HLS, or Microsoft Smooth Streaming have explicit support for requesting just the portion of a stream needed at the moment. Even more traditional HTTP downloads can be separated into chunks. Since it is likely that at some point in the future the user will be requesting the next set of bytes in the stream or the file, prefetching attempts to load that data into the edge server's cache before it is requested by the user. When it is successful, prefetching reduces latency to the user as the next set of data will have already been transferred from the origin to the edge server by the time the user asks for it.

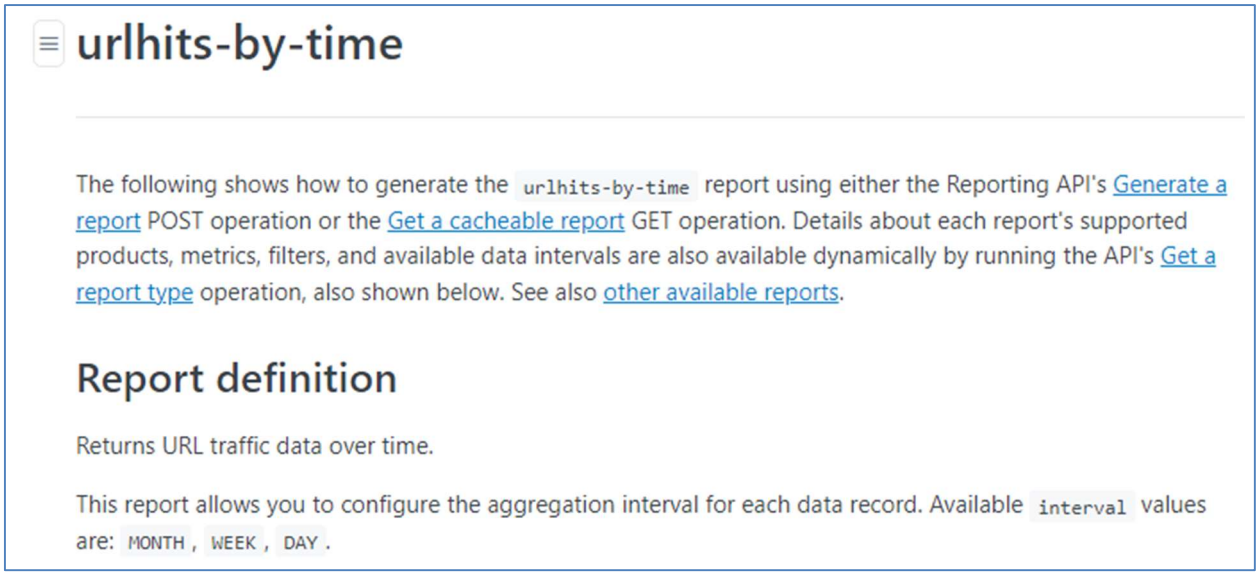
See e.g., https://developer.akamai.com/article/prefetching.



serve their objects without having to send requests to the origin. As the cache on a server fills up, our software looks for the least recently used objects in its cache and evicts them to make room for new objects. Objects that are requested most often will remain in cache for a long time, while less requested objects will be automatically purged as requests for new objects arrive. Content is never evicted unless there are requests for new objects and there is no existing room in the cache for them, or unless the customer explicitly issues a purge request to have an object removed from the network.

For websites with a particularly high volume, or during request surges, Akamai offers a Tiered Distribution method (AKA cache hierarchy). Tiered Distribution creates "layers" of cached objects, depending on access to speed and frequency of object request. The first layer of a cache hierarchy is the highest speed and is made up of the most requested objects. When that is filled, a second layer, which is slightly slower will be created with the second most requested items, and so on. This allows for reduced latency and highly efficient delivery from caching.

See e.g., https://developer.akamai.com/article/content-caching.



urlhits-by-time

The following shows how to generate the `urlhits-by-time` report using either the Reporting API's Generate a report POST operation or the Get a cacheable report GET operation. Details about each report's supported products, metrics, filters, and available data intervals are also available dynamically by running the API's Get a report type operation, also shown below. See also other available reports.

Report definition

Returns URL traffic data over time.

This report allows you to configure the aggregation interval for each data record. Available `interval` values are: `MONTH`, `WEEK`, `DAY`.

See e.g., https://techdocs.akamai.com/reporting/reference/urlhits-by-time.

Standard dimensions

Media Analytics provides a number of standard dimensions that are based on data that the system inherently gathers from log data and Akamai's EdgeScape service.

Dimension information levels

Before you apply dimensions, you should understand how dimension levels are applied in Media Analytics.

- **Viewer level**. These dimensions refer to a viewer's current session and all future sessions. Dimensions at the viewer level are also available with Visit and Play metrics, whereas metrics at the viewer level are available only with viewer level dimensions. You will have to restrict the number of distinct values per dimension to 5000000.
- **Visit level**. These dimensions refer to viewer playback activity in a certain duration with one or more titles played during a visit. They can also be used with Play metrics.
- **Play level**. These dimensions refer to a single title playback event. Attempts to play back video that resulted in errors are also included. These dimensions can only be used with Play level metrics.

The dimensions

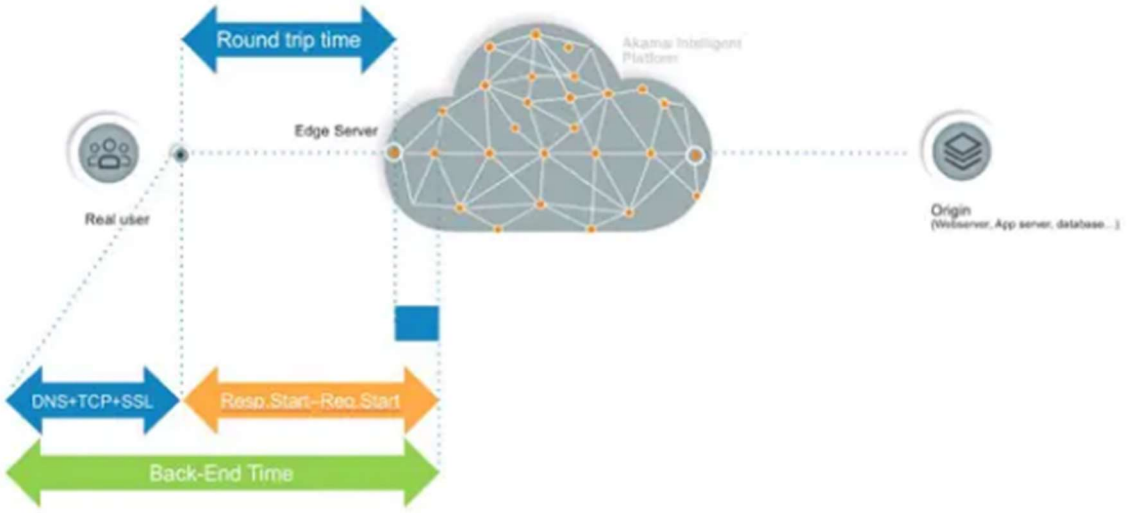
The following standard dimensions are supported:

Dimension	Definition
ASNUM	Autonomous System Number. This identifies the internet network.
CDN	Content Distribution Network. This identifies the CDN used to deliver the content. For example, <>.
Connection Speed	The internet connection speed of the viewer. Available values are: • 1 to 55 Kbps: BW = 1 • 56 to 255 Kbps: BW = 56 • 256 to 999 Kbps: BW = 256 • 1000 to 1999 Kbps: BW = 1000 • 2000 to 4999 Kbps: BW = 2000 • 5000 Kbps+ : BW = 5000
Connection Type	The internet connection type. For example, Wi-Fi or cellular.
DMA	Designated Market Area.
Day of week	Day as per the time zone specified in the Analyzer.

Event Name	Name of the event.
Event URL	Stream URL.
Format	The format in which the media is delivered. Available values are: • WMS • Flash • Progressive Download
Geography	Viewer location. Available granularities are: • Continent • Region • Country • City (available for the US and Canada only)
Time	Time as per the time zone specified in the Analyzer.
Referer	URL of the webpage that refers to the requested media.

See e.g., https://techdocs.akamai.com/media-analytics/docs/standard-dimensions.

2. **Cache hit example**: When there is a cache hit (i.e., content is served by an edge server), the request will *not* go all the way back to the origin, as shown in the diagram below.



See e.g., https://developer.akamai.com/blog/2018/11/16/using-cdn-server-timing-monitor-cdn-and-origin-performance.

After downloading an object the Akamai servers will save a copy of it. Servers within the same deployment are able to check each other's caches using the Inter-Cache Protocol (ICP). This means that the amount of storage space for caching content is the sum of the storage space of all the servers in that deployment.

When our DNS servers reply to lookup requests, we generally return two IP addresses for any given deployment. Both servers will cache the objects that they serve, so the total cache space is half the total storage space of that deployment. This redundancy allows us to take machines out of service for installs, or maintenance, or due to an outage, and still be able to serve their objects without having to send requests to the origin. As the cache on a server fills up, our software looks for the least recently used objects in its cache and evicts them to make room for new objects. Objects that are requested most often will remain in cache for a long time, while less requested objects will be automatically purged as requests for new objects arrive. Content is never evicted unless there are requests for new objects and there is no existing room in the cache for them, or unless the customer explicitly issues a purge request to have an object removed from the network.

For websites with a particularly high volume, or during request surges, Akamai offers a Tiered Distribution method (AKA cache hierarchy). Tiered Distribution creates "layers" of cached objects, depending on access to speed and frequency of object request. The first layer of a cache hierarchy is the highest speed and is made up of the most requested objects. When that is filled, a second layer, which is slightly slower will be created with the second most requested items, and so on. This allows for reduced latency and highly efficient delivery from caching.

See e.g., https://developer.akamai.com/article/content-caching.

133. On information and belief, one or more components of the Roku system employ and provide a method comprising the storing of the pre-fetched contents.

serve their objects without having to send requests to the origin. As the cache on a server fills up, our software looks for the least recently used objects in its cache and evicts them to make room for new objects. Objects that are requested most often will remain in cache for a long time, while less requested objects will be automatically purged as requests for new objects arrive. Content is never evicted unless there are requests for new objects and there is no existing room in the cache for them, or unless the customer explicitly issues a purge request to have an object removed from the network.

For websites with a particularly high volume, or during request surges, Akamai offers a Tiered Distribution method (AKA cache hierarchy). Tiered Distribution creates "layers" of cached objects, depending on access to speed and frequency of object request. The first layer of a cache hierarchy is the highest speed and is made up of the most requested objects. When that is filled, a second layer, which is slightly slower will be created with the second most requested items, and so on. This allows for reduced latency and highly efficient delivery from caching.

See e.g., https://developer.akamai.com/article/content-caching.

What is a CDN (Content Delivery Network)?

A content delivery network (CDN) is a group of geographically distributed servers that speed up the delivery of web content by bringing it closer to where users are. Data centers across the globe use caching, a process that temporarily stores copies of files, so that you can access internet content from a web-enabled device or browser more quickly through a server near you. CDNs cache content like web pages, images, and video in proxy servers near to your physical location. This allows you to do things like watch a movie, download software, check your bank balance, post on social media, or make purchases, without having to wait for content to load.

See e.g., https://www.akamai.com/our-thinking/cdn/what-is-a-cdn.

What is an example of a CDN?

A large portion of all internet content is delivered through CDNs. Here is a simple example:

If you were in New York and wanted to view the website of your favorite store in London that's hosted on a server in the UK, you would experience slow content load times if the request had to travel all the way across the Atlantic Ocean. To remedy this, a CDN would store a cached version of the London website content in multiple geographical locations around the world, also called "points of presence" (PoPs). These PoPs contain their own caching servers and are responsible for delivering that content close to where you're located in New York.

Content delivered from a server closest to your physical location gives you a faster, high-performance web experience.

See e.g., https://www.akamai.com/our-thinking/cdn/what-is-a-cdn.

Key Features

1. **IP Anycast**: Provide a decentralized DNS service to users that enables the creation of a logical name server comprising multiple physical servers deployed across multiple networks and continents.

See e.g., https://developer.akamai.com/edge-dns

How does a CDN work?

The mission of a CDN is to reduce *latency*. Latency is that annoying delay you experience when trying to access a web page or video stream before it fully loads on your device. Although measured in milliseconds, it can feel like forever, and may even result in a load error or time-out. Some content delivery networks alleviate latency by reducing the physical distance that the content needs to travel to reach you. Therefore, larger, more widely distributed CDNs are able to deliver web content more quickly and reliably by putting the content as close to the end user as possible.

Let's say it's the weekend and you want to kick back and stream the latest Hollywood movie release — the CDN finds an optimal server on its network to serve up that video. Usually, that will be the server closest to your physical location. The media files will be cached and remain on that content delivery network server for other user requests in the same geographic area. If the content you requested is unavailable or outdated, the CDN service will store the newly fetched content to serve any future requests.

See e.g., https://www.akamai.com/our-thinking/cdn/what-is-a-cdn.

After downloading an object the Akamai servers will save a copy of it. Servers within the same deployment are able to check each other's caches using the Inter-Cache Protocol (ICP). This means that the amount of storage space for caching content is the sum of the storage space of all the servers in that deployment.

When our DNS servers reply to lookup requests, we generally return two IP addresses for any given deployment. Both servers will cache the objects that they serve, so the total cache space is half the total storage space of that deployment. This redundancy allows us to take machines out of service for installs, or maintenance, or due to an outage, and still be able to serve their objects without having to send requests to the origin. As the cache on a server fills up, our software looks for the least recently used objects in its cache and evicts them to make room for new objects. Objects that are requested most often will remain in cache for a long time, while less requested objects will be automatically purged as requests for new objects arrive. Content is never evicted unless there are requests for new objects and there is no existing room in the cache for them, or unless the customer explicitly issues a purge request to have an object removed from the network.

For websites with a particularly high volume, or during request surges, Akamai offers a Tiered Distribution method (AKA cache hierarchy). Tiered Distribution creates "layers" of cached objects, depending on access to speed and frequency of object request. The first layer of a cache hierarchy is the highest speed and is made up of the most requested objects. When that is filled, a second layer, which is slightly slower will be created with the second most requested items, and so on. This allows for reduced latency and highly efficient delivery from caching.

See e.g., https://developer.akamai.com/article/content-caching.

134. On information and belief, one or more components of the Roku system employ and provide a method comprising the delivering of the stored more-frequently-accessed contents to at least one of the plurality of devices for accessing content.

serve their objects without having to send requests to the origin. As the cache on a server fills up, our software looks for the least recently used objects in its cache and evicts them to make room for new objects. Objects that are requested most often will remain in cache for a long time, while less requested objects will be automatically purged as requests for new objects arrive. Content is never evicted unless there are requests for new objects and there is no existing room in the cache for them, or unless the customer explicitly issues a purge request to have an object removed from the network.

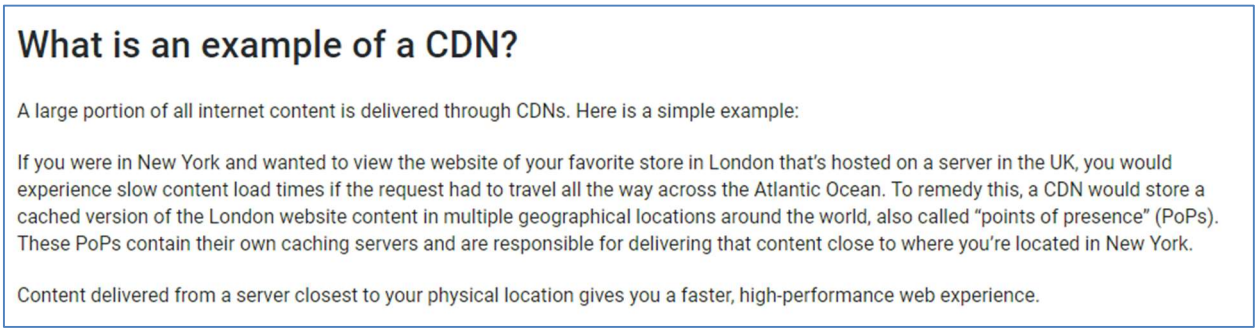
For websites with a particularly high volume, or during request surges, Akamai offers a Tiered Distribution method (AKA cache hierarchy). Tiered Distribution creates "layers" of cached objects, depending on access to speed and frequency of object request. The first layer of a cache hierarchy is the highest speed and is made up of the most requested objects. When that is filled, a second layer, which is slightly slower will be created with the second most requested items, and so on. This allows for reduced latency and highly efficient delivery from caching.

See e.g., https://developer.akamai.com/article/content-caching.

What is a CDN (Content Delivery Network)?

A content delivery network (CDN) is a group of geographically distributed servers that speed up the delivery of web content by bringing it closer to where users are. Data centers across the globe use caching, a process that temporarily stores copies of files, so that you can access internet content from a web-enabled device or browser more quickly through a server near you. CDNs cache content like web pages, images, and video in proxy servers near to your physical location. This allows you to do things like watch a movie, download software, check your bank balance, post on social media, or make purchases, without having to wait for content to load.

See e.g., https://www.akamai.com/our-thinking/cdn/what-is-a-cdn.



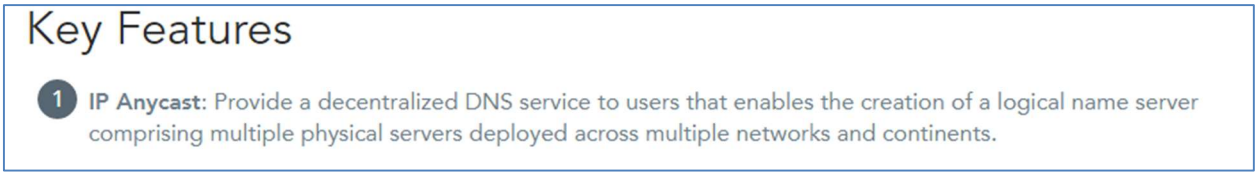
What is an example of a CDN?

A large portion of all internet content is delivered through CDNs. Here is a simple example:

If you were in New York and wanted to view the website of your favorite store in London that's hosted on a server in the UK, you would experience slow content load times if the request had to travel all the way across the Atlantic Ocean. To remedy this, a CDN would store a cached version of the London website content in multiple geographical locations around the world, also called "points of presence" (PoPs). These PoPs contain their own caching servers and are responsible for delivering that content close to where you're located in New York.

Content delivered from a server closest to your physical location gives you a faster, high-performance web experience.

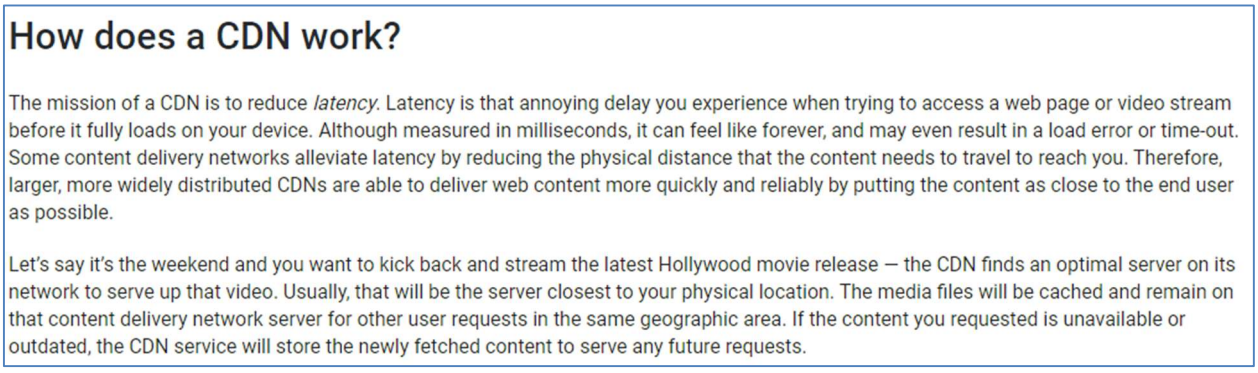
See e.g., https://www.akamai.com/our-thinking/cdn/what-is-a-cdn.



Key Features

1 **IP Anycast**: Provide a decentralized DNS service to users that enables the creation of a logical name server comprising multiple physical servers deployed across multiple networks and continents.

See e.g., https://developer.akamai.com/edge-dns

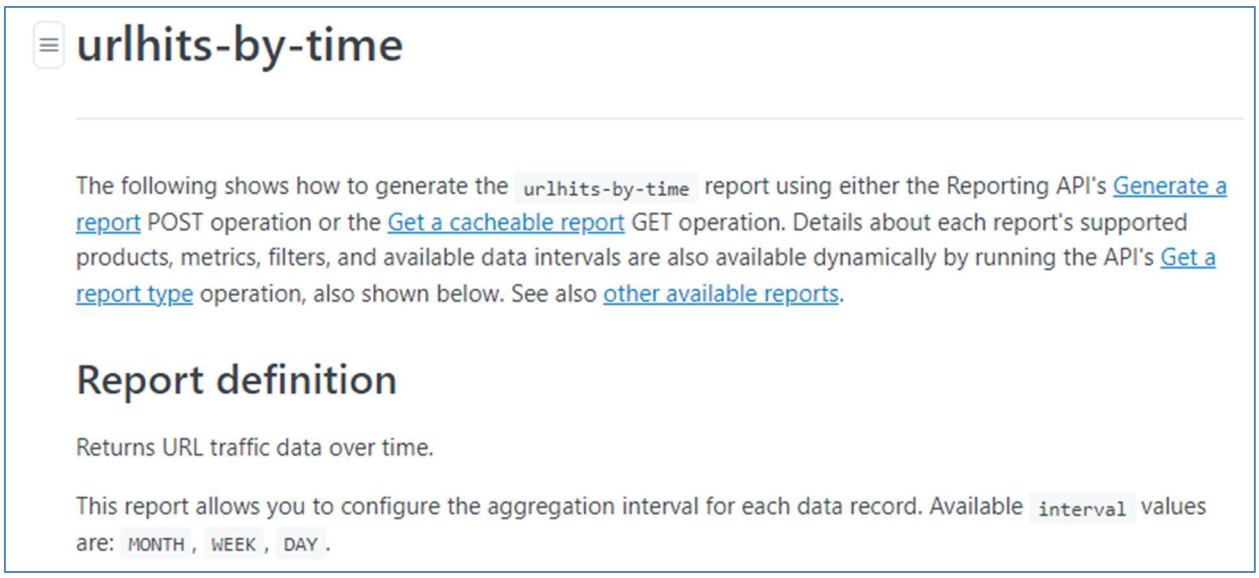


How does a CDN work?

The mission of a CDN is to reduce *latency*. Latency is that annoying delay you experience when trying to access a web page or video stream before it fully loads on your device. Although measured in milliseconds, it can feel like forever, and may even result in a load error or time-out. Some content delivery networks alleviate latency by reducing the physical distance that the content needs to travel to reach you. Therefore, larger, more widely distributed CDNs are able to deliver web content more quickly and reliably by putting the content as close to the end user as possible.

Let's say it's the weekend and you want to kick back and stream the latest Hollywood movie release — the CDN finds an optimal server on its network to serve up that video. Usually, that will be the server closest to your physical location. The media files will be cached and remain on that content delivery network server for other user requests in the same geographic area. If the content you requested is unavailable or outdated, the CDN service will store the newly fetched content to serve any future requests.

See e.g., https://www.akamai.com/our-thinking/cdn/what-is-a-cdn.



urlhits-by-time

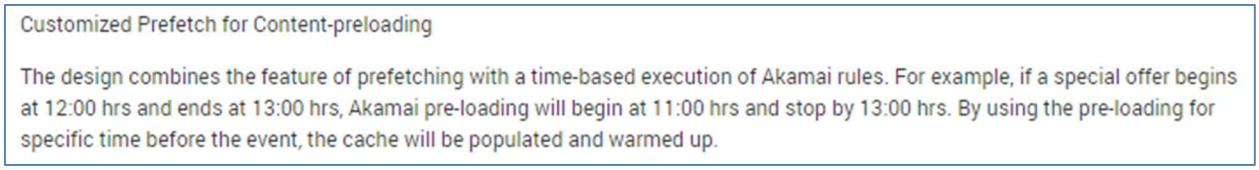
The following shows how to generate the `urlhits-by-time` report using either the Reporting API's Generate a report POST operation or the Get a cacheable report GET operation. Details about each report's supported products, metrics, filters, and available data intervals are also available dynamically by running the API's Get a report type operation, also shown below. See also other available reports.

Report definition

Returns URL traffic data over time.

This report allows you to configure the aggregation interval for each data record. Available `interval` values are: `MONTH`, `WEEK`, `DAY`.

See e.g., https://techdocs.akamai.com/reporting/reference/urlhits-by-time.



Customized Prefetch for Content-preloading

The design combines the feature of prefetching with a time-based execution of Akamai rules. For example, if a special offer begins at 12:00 hrs and ends at 13:00 hrs, Akamai pre-loading will begin at 11:00 hrs and stop by 13:00 hrs. By using the pre-loading for specific time before the event, the cache will be populated and warmed up.

See e.g., https://community.akamai.com/customers/s/article/Content-Pre-loading-pre-warming-at-Akamai?language=en_US.

135. On information and belief, one or more components of the Roku system employ and provide a method comprising periodically updating the stored pre-fetched content, where the updating is performed according to the collected time-stamped data of the stored pre-fetched contents and comprises pre-fetching the more-frequently-accessed contents according to their control data and storing said pre-fetched contents so that said pre-fetched content is updated timely and is delivered to at least one of the plurality of devices for accessing content before further selection by the user.

Setting Time-to-live

Time-to-live (TTL) is the amount of time that an object can be served from the cache. Specifically for Akamai, TTL is the time period an Akamai Edge server may serve an object from the cache without revalidating its freshness with the origin server.

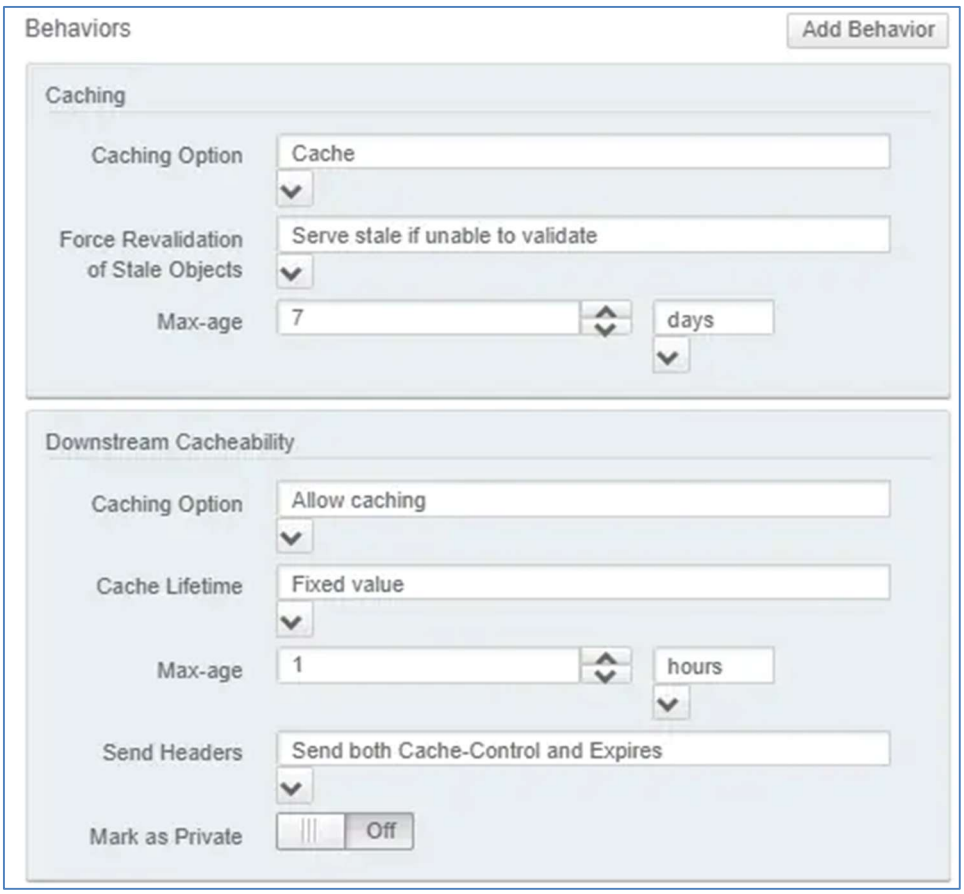
As a guideline, for Akamai caching, if the TTL is short, you're not achieving maximum offload of your origin servers. However, if the TTL is long, you have to purge web content all the time, unless you have file versioning coding best practices. Furthermore, browser caching TTL should be less than the Akamai caching TTL — be careful with these settings, as you can't clear your customer's browser cache.

* * *

Implementing Delivery Configurations

As mentioned above, to avoid caching PII data, please make sure your Akamai delivery configuration contains a No Store caching behavior in the Default Rule. Rules are processed top-down in Akamai delivery configurations, so be careful that the caching rules towards the bottom of the delivery configuration don't override those towards the top. For example, if you have a rule towards the top for images with a TTL of 1 hour, and another rule below it of 7 days for the same images, the 7-day rule would take precedence.

* * *



See e.g., https://developer.akamai.com/blog/2020/06/09/introduction-caching-akamai.

Cache lifetime options

The following table lists the cache lifetime options that you can choose after selecting the **Allow caching** downstream caching option. The cache lifetime is determined based on the `Expires` header; each option listed below sets the `Expires` header to a specific value.

Name	Description
Smaller value: origin header or remaining edge TTL	Sets caching lifetime to a value that is lesser of the remaining edge cache lifetime and origin header value (`Cache-Control:max-age` or `Expires` , if both are present, `Cache-Control:max-age` takes precedence).
Greater value: origin header or remaining edge TTL	Sets caching lifetime to a value that is greater of the remaining edge cache lifetime and origin header value (`Cache-Control:max-age` or `Expires` , if both are present, `Cache-Control:max-age` takes precedence).
Remaining edge TTL	Sets caching lifetime to the remaining edge cache lifetime.
Full edge TTL (max-age)	Sets caching lifetime to the full edge cache lifetime as represented by the `max-age` parameter.
Fixed value	Sets caching lifetime to a fixed value that you can define.
Calculate Expires from origin Cache-Control	Sets caching lifetime to the origin's `Cache-Control:max-age` header value.

See e.g., https://techdocs.akamai.com/api-definitions/docs/caching.

serve their objects without having to send requests to the origin. As the cache on a server fills up, our software looks for the least recently used objects in its cache and evicts them to make room for new objects. Objects that are requested most often will remain in cache for a long time, while less requested objects will be automatically purged as requests for new objects arrive. Content is never evicted unless there are requests for new objects and there is no existing room in the cache for them, or unless the customer explicitly issues a purge request to have an object removed from the network.

For websites with a particularly high volume, or during request surges, Akamai offers a Tiered Distribution method (AKA cache hierarchy). Tiered Distribution creates "layers" of cached objects, depending on access to speed and frequency of object request. The first layer of a cache hierarchy is the highest speed and is made up of the most requested objects. When that is filled, a second layer, which is slightly slower will be created with the second most requested items, and so on. This allows for reduced latency and highly efficient delivery from caching.

See e.g., https://developer.akamai.com/article/content-caching.

What is a CDN (Content Delivery Network)?

A content delivery network (CDN) is a group of geographically distributed servers that speed up the delivery of web content by bringing it closer to where users are. Data centers across the globe use caching, a process that temporarily stores copies of files, so that you can access internet content from a web-enabled device or browser more quickly through a server near you. CDNs cache content like web pages, images, and video in proxy servers near to your physical location. This allows you to do things like watch a movie, download software, check your bank balance, post on social media, or make purchases, without having to wait for content to load.

See e.g., https://www.akamai.com/our-thinking/cdn/what-is-a-cdn.

What is an example of a CDN?

A large portion of all internet content is delivered through CDNs. Here is a simple example:

If you were in New York and wanted to view the website of your favorite store in London that's hosted on a server in the UK, you would experience slow content load times if the request had to travel all the way across the Atlantic Ocean. To remedy this, a CDN would store a cached version of the London website content in multiple geographical locations around the world, also called "points of presence" (PoPs). These PoPs contain their own caching servers and are responsible for delivering that content close to where you're located in New York.

Content delivered from a server closest to your physical location gives you a faster, high-performance web experience.

See e.g., https://www.akamai.com/our-thinking/cdn/what-is-a-cdn.

Key Features

1. **IP Anycast**: Provide a decentralized DNS service to users that enables the creation of a logical name server comprising multiple physical servers deployed across multiple networks and continents.

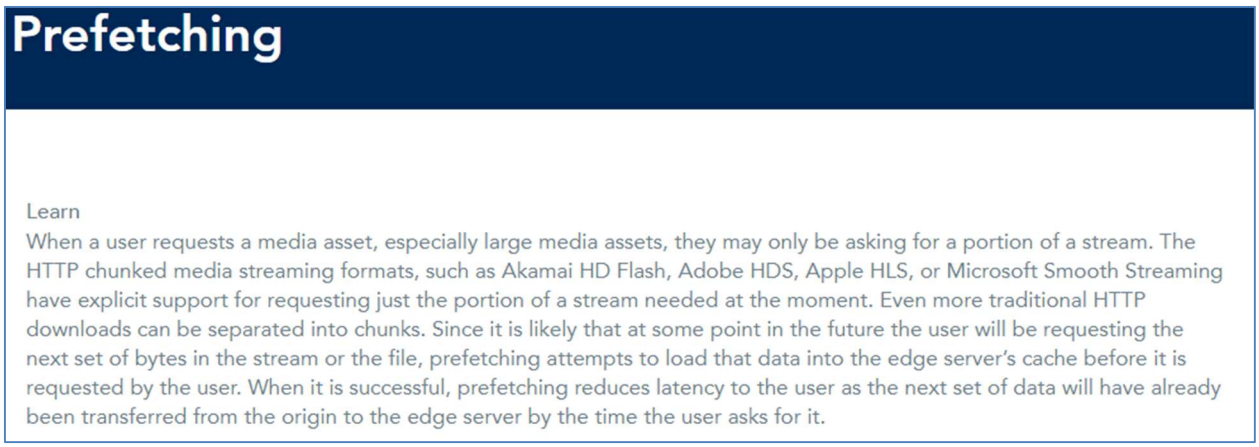
See e.g., https://developer.akamai.com/edge-dns.

How does a CDN work?

The mission of a CDN is to reduce *latency*. Latency is that annoying delay you experience when trying to access a web page or video stream before it fully loads on your device. Although measured in milliseconds, it can feel like forever, and may even result in a load error or time-out. Some content delivery networks alleviate latency by reducing the physical distance that the content needs to travel to reach you. Therefore, larger, more widely distributed CDNs are able to deliver web content more quickly and reliably by putting the content as close to the end user as possible.

Let's say it's the weekend and you want to kick back and stream the latest Hollywood movie release — the CDN finds an optimal server on its network to serve up that video. Usually, that will be the server closest to your physical location. The media files will be cached and remain on that content delivery network server for other user requests in the same geographic area. If the content you requested is unavailable or outdated, the CDN service will store the newly fetched content to serve any future requests.

See e.g., https://www.akamai.com/our-thinking/cdn/what-is-a-cdn.



See e.g., https://developer.akamai.com/article/prefetching.

136. Roku's direct infringement has damaged AdaptFlow and caused it to suffer and continue to suffer irreparable harm and damages.

JURY DEMANDED

137. Pursuant to Federal Rule of Civil Procedure 38(b), AdaptFlow hereby requests a trial by jury on all issues so triable.

PRAYER FOR RELIEF

AdaptFlow respectfully requests this Court to enter judgment in Advanced Transaction's favor and against Roku as follows:

a. finding that Roku has infringed one or more claims of the '034 patent under 35 U.S.C. §§ 271(a);

b. finding that Roku has infringed one or more claims of the '494 patent under 35 U.S.C. §§ 271(a);

c. finding that Roku has infringed one or more claims of the '528 patent under 35 U.S.C. §§ 271(a);

d. finding that Roku has infringed one or more claims of the ’012 patent under 35 U.S.C. §§ 271(a);

e. finding that Roku has infringed one or more claims of the ’162 patent under 35 U.S.C. §§ 271(a);

f. finding that Roku has infringed one or more claims of the ’068 patent under 35 U.S.C. §§ 271(a);

g. finding that Roku has infringed one or more claims of the ’757 patent under 35 U.S.C. §§ 271(a);

h. finding that Roku has infringed one or more claims of the ’064 patent under 35 U.S.C. §§ 271(a);

i. awarding AdaptFlow damages under 35 U.S.C. § 284, or otherwise permitted by law, including supplemental damages for any continued post-verdict infringement;

j. awarding AdaptFlow pre-judgment and post-judgment interest on the damages award and costs;

k. awarding cost of this action (including all disbursements) and attorney fees pursuant to 35 U.S.C. § 285, or as otherwise permitted by the law; and

l. awarding such other costs and further relief that the Court determines to be just and equitable.

Dated: September 22, 2022

Respectfully submitted,

/s/ Raymond W. Mort, III

Raymond W. Mort, III
Texas State Bar No. 00791308
raymort@austinlaw.com

THE MORT LAW FIRM, PLLC
501 Congress Avenue, Suite 150
Austin, Texas 78701
Tel/Fax: 512-865-7343

Of Counsel:
Ronald M. Daignault (*pro hac vice* to be filed)*
Chandran B. Iyer (*pro hac vice* to be filed)
Oded Burger (*pro hac vice* to be filed)*
Tedd W. Van Buskirk (*pro hac vice* to be filed)*
Zachary H. Ellis (Texas State Bar No. 24122606)*
rdaignault@daignaultiyer.com
cbiyer@daignaultiyer.com
oburger@daignaultiyer.com
tvanbuskirk@daignaultiyer.com
zellis@daignaultiyer.com
DAIGNAULT IYER LLP
8618 Westwood Center Drive - Suite 150
Vienna, VA 22182

Attorneys for Plaintiff AdaptFlow Technologies LLC.

*Not admitted to practice in Virginia